Summarizing Information Graphics Textually

Seniz Demir*
TUBITAK-BILGEM

Sandra Carberry**
University of Delaware

Kathleen F. McCoy†
University of Delaware

Information graphics (such as bar charts and line graphs) play a vital role in many multimodal documents. The majority of information graphics that appear in popular media are intended to convey a message and the graphic designer uses deliberate communicative signals, such as highlighting certain aspects of the graphic, in order to bring that message out. The graphic, whose communicative goal (intended message) is often not captured by the document's accompanying text, contributes to the overall purpose of the document and cannot be ignored. This article presents our approach to providing the high-level content of a non-scientific information graphic via a brief textual summary which includes the intended message and the salient features of the graphic. This work brings together insights obtained from empirical studies in order to determine what should be contained in the summaries of this form of non-linguistic input data, and how the information required for realizing the selected content can be extracted from the visual image and the textual components of the graphic. This work also presents a novel bottom–up generation approach to simultaneously construct the discourse and sentence structures of textual summaries by leveraging different discourse related considerations such as the syntactic complexity of realized sentences and clause embeddings. The effectiveness of our work was validated by different evaluation studies.

1. Introduction

Graphical representations are widely used to depict quantitative data and the relations among them (Friendly 2008). Although some graphics are constructed from raw data only for visualization purposes, the majority of information graphics (such as bar charts and line graphs) found in popular media (such as magazines and newspapers) are

* The Scientific and Technological Research Council of Turkey, Center of Research for Advanced Technologies of Informatics and Information Security, Gebze, Kocaeli, TURKEY, 41470. E-mail: senizd@uekae.tubitak.gov.tr. (This work was done while the author was a graduate student at the Department of Computer and Information Sciences, University of Delaware, Newark, DE, USA 19716.)
** Department of Computer and Information Sciences, University of Delaware, Newark, DE, USA 19716. E-mail: carberry@cis.udel.edu.
† Department of Computer and Information Sciences, University of Delaware, Newark, DE, USA 19716. E-mail: mccoy@cis.udel.edu.

Submission received: 20 April 2010; revised submission received: 8 July 2011; accepted for publication: 6 September 2011.

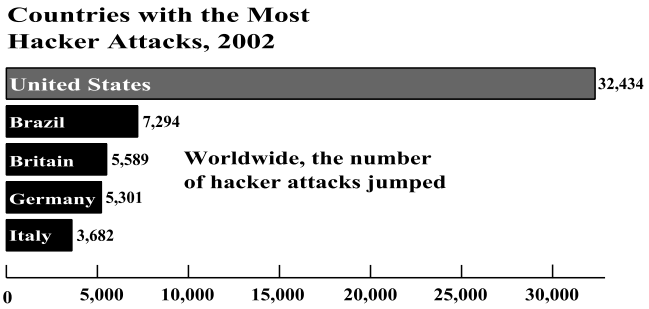


Figure 1
Graphic conveying a maximum bar.

constructed to convey a message. For example, the graphic in Figure 1 ostensibly is intended to convey that "The United States has the highest number of hacker attacks among the countries listed." The graphic designer made deliberate choices in order to bring that message out. For example, the bar representing the United States is highlighted with a different color from the other bars and the bars are sorted with respect to their values instead of their labels so that the bar with the highest value can be easily recognized. Such choices, we argue, are examples of communicative signals that graphic designers use. Under Clark's definition (1996), language is not just text and utterances, but instead includes any deliberate signal (such as gestures and facial expressions) that is intended to convey a message; thus an information graphic is a form of language.

In popular media, information graphics often appear as part of a multimodal document. Carberry, Elzer, and Demir (2006) conducted a corpus study of information graphics from popular media, where the extent to which the message of a graphic is also captured by the text of the accompanying document was analyzed. One hundred randomly selected graphics of different kinds (e.g., bar charts and line graphs) were collected from newspapers and magazines along with their articles. It was observed that in 26% of the instances, the text conveyed only a small portion of the graphic's message and in 35% of the instances, the text didn't capture the graphic's message at all. Thus graphics, together with the textual segments, contribute to the overall purpose of a document (Grosz and Sidner 1986) and cannot be ignored. We argue that information graphics are an important knowledge resource that should be exploited, and understanding the intention of a graphic is the first step towards exploiting it.

This article presents our novel approach to identifying and textually conveying the high-level content of an information graphic (the message and knowledge that one would gain from viewing a graphic) from popular media. Our system summarizes this form of non-linguistic input data by utilizing the inferred intention of the graphic designer and the communicative signals present in the visual representation. Our overall goal is to generate a succinct coherent summary of a graphic that captures the intended message of the graphic and its visually salient features, which we hypothesize as being related to the intended message. Input to our system is the intention of the graphic inferred by the Bayesian Inference System (Elzer, Carberry, and Zukerman 2011), and an XML representation of the visual graphic (Chester and Elzer 2005) that specifies the components of the graphic such as the number of bars and the heights of each bar. Our work focuses on the generation issues inherent in generating a textual summary of a graphic given this information. The current implementation of the system is applicable to only one kind of information graphic, simple bar charts, but we hypothesize that the overall summarization approach could be extended to other kinds of graphics.

In this article, we investigate answers to the following questions: (1) Among all possible information that could be conveyed about a bar chart, what should be included in its summary? (2) How should the content of a summary be organized into a coherent text? (3) How should the text structure be best realized in natural language? Given the intended message and the XML representation of a graphic, our system first determines the content of the graphic's summary (a list of propositions) by applying the content identification rules constructed for that intended message category. Our system then produces a coherent organization of the selected content by applying a bottom–up approach which leverages a variety of considerations (such as the syntactic complexity of the realized sentences and clause embeddings) in choosing how to aggregate information into sentence-sized units. The system finally orders and realizes the sentence-sized units in natural language and generates referring expressions for graphical elements that are required in realization.

The rest of this article is structured as follows. Section 2 discusses related work on summarization of non-linguistic input data and describes some natural language applications which could benefit from summaries generated by our work. Section 3 outlines our summarization framework. Section 4 is concerned with identifying the propositional content of a summary and presents our content-identification rules that specify what should be included in the summary of a graphic. Section 5 describes our bottom–up approach, which applies operators to relate propositions selected for inclusion, explores aggregating them into sentence-sized units, and selects the best organization via an evaluation metric. Section 6 presents our sentence-ordering mechanism, which incorporates centering theory to specify the order in which the sentence-sized units should be presented. Section 7 describes how our system realizes the selected content in natural language. Particular attention is devoted to our methodology for generating referring expressions for certain graphical elements such as a descriptor of what is being measured in the graphic. Section 8 presents a user study that was conducted to evaluate the effectiveness of the generated summaries for the purposes of this research by measuring readers' comprehension. Section 9 concludes the article and outlines our future work.

2. Background

2.1 Related Work

There has been a growing interest in language systems that generate textual summaries of non-linguistic input data (Reiter 2007). The overall goal of these systems, generally referred to as **data-to-text systems**, is to enable efficient processing of large volumes of numeric data by supporting traditional visualisation modalities and to reduce the effort spent by human experts on analyzing the data. Various examples of data-to-text systems in the literature include systems that summarize weather forecast data (Goldberg, Driedger, and Kittredge 1994; Coch 1998), stock market data (Kukich 1983), and georeferenced data (Turner, Sripada, and Reiter 2009).

One of the most successful data-to-text generation research efforts is the SumTime project, which uses pattern recognition techniques to generate textual summaries of automatically generated time-series data in order to convey the significant and interesting events (such as spikes and oscillations) that a domain expert would recognize by analyzing the data. The SumTime-Mousam (Somayajulu, Reiter, and Davy 2003) and SumTime-Turbine (Yu et al. 2007) systems were designed to summarize weather forecast data and the data from gas turbine engines, respectively. More recently, the

project was extended to the medical domain. The BabyTalk (Gatt et al. 2009) project produces textual summaries of clinical data collected for babies in a neonatal intensive care unit, where the summaries are intended to present key information to medical staff for decision support. The implemented prototype (BT-45) (Portet et al. 2009) generates multi-paragraph summaries from large quantities of heterogeneous data (e.g., time series sensor data and the records of actions taken by the medical staff). The overall goal of these systems (identifying and presenting significant events) is similar to our goal of generating a summary that conveys what a person would get by viewing an information graphic, and these systems contend with each of the generation issues we must face with our system. Our generation methodology, however, is different from the approaches deployed in these systems in various respects. For example, BT-45 produces multi-paragraph summaries where each paragraph presents first a key event (of highest importance), then events related to the key event (e.g., an event that causes the key event), and finally other co-temporal events. Our system, on the other hand, produces single-paragraph summaries where the selected propositions are grouped and ordered with respect to the kind of information they convey. In addition, BT-45 performs a limited amount of aggregation at the conceptual level, where the aggregation is used to express the relations between events with the use of temporal adverbials and cue phrases (such as *as a result*). Contrarily, our system syntactically aggregates the selected propositions with respect to the entities they share.

There is also a growing literature on summarizing numeric data visualized via graphical representations. One of the recent studies, the iGRAPH-Lite (Ferres et al. 2007) system, provides visually impaired users access to the information in a graphic via keyboard commands. The system is specifically designed for the graphics that appear in "The Daily" (Statistics Canada's main dissemination venue) and presents the user with a template-based textual summary of the graphic. Although this system is very useful for in depth analysis of statistical graphs and interpreting numeric data, it is not appropriate for graphics from popular media where the intended message of the graphic is important. In the iGRAPH-Lite system, the summary generated for a graphic conveys the same information (such as the title of the graphic, and the maximum and minimum values) no matter what the visual features of the graphic are. The content of the summaries that our system generates, however, is dependent on the intention and the visual features of the graphic. Moreover, that system does not consider many of the generation issues that we address in our work.

Choosing an appropriate presentation for a large amount of quantitative data is a difficult and time-consuming task (Foster 1999). A variety of systems were built to automatically generate presentations of statistical data—such as the PostGraphe system (Corio and Lapalme 1999; Fasciano and Lapalme 2000), which generates graphics and complementary text based on the information explicitly given by the user such as the intention to be conveyed in the graphic and the data of special interest to the user. The content of the accompanying text is determined according to the intention of the graphic and the features of the data. Moreover, the generated texts are intended to reinforce some important facts that are visually present in the graphic. In this respect, the generation in PostGraphe is similar to our work, although the output texts have a limited range and are heavily dependent on the information explicitly given by the user.

2.2 Role of Graphical Summaries in Natural Language Applications

2.2.1 Accessibility. Electronic documents that contain information graphics pose challenging problems for visually impaired individuals. The information residing in the

text can be delivered via screen reader programs but visually impaired individuals are generally stymied when they come across graphics. These individuals can only receive the ALT text (human-generated text that conveys the content of a graphic) associated with the graphic. Many electronic documents do not provide ALT texts and even in the cases where ALT text is present, it is often very general or inadequate for conveying the intended message of the graphic (Lazar, Kleinman, and Malarkey 2007).

Researchers have explored different techniques for providing access to the informational content of graphics for visually impaired users, such as sound (Meijer 1992; Alty and Rigas 1998), touch (Ina 1996; Jayant et al. 2007), or a combination of the two (Kennel 1996; Ramloll et al. 2000). Unfortunately, these approaches have serious limitations such as requiring the use of special equipment (e.g., printers and touch panels) or preparation work done by sighted individuals. Research has also investigated language-based accessibility systems to provide access to graphics (Kurze 1995; Ferres et al. 2007). As mentioned in Section 2.1, these language-based systems are not appropriate for graphics in articles from popular media where the intended message of the graphic is important. We hypothesize that providing alternative access to what the graphic looks like is not enough and that the user should be provided with the message and knowledge that one would gain from viewing the graphic. We argue that the textual summaries generated by our approach could be associated with graphics as ALT texts so that individuals with sight impairments would be provided with the high-level content of graphics while reading electronic documents via screen readers.

2.2.2 Document Summarization. Research has extensively investigated various techniques for single (Hovy and Lin 1996; Baldwin and Morton 1998) and multi-document summarization (Goldstein et al. 2000; Schiffman, Nenkova, and McKeown 2002). The summary should provide the topic and an overview of the summarized documents by identifying the important and interesting aspects of these documents. Document summarizers generally evaluate and extract items of information from documents according to their relevance to a particular request (such as a request for a person or an event) and address discourse related issues such as removing redundancies (Radev et al. 2004) and ordering sentences (Barzilay, Elhadad, and McKeown 2002) in order to make the summary more coherent.

It is widely accepted that to produce a good summary of a document, one must understand the document and recognize the communicative intentions of the author. Summarization work primarily focuses on the text of a document but, as mentioned earlier, information graphics are an important part of many multimodal documents that appear in popular media and these graphics contribute to the overall communicative intention of the document. We argue that document summarization should capture the high-level content of graphics that are included in the document, because information graphics often convey information that is not repeated elsewhere in the document. We believe that the summary of a graphic generated by our system, which provides the intended message of the graphic and the information that would be perceived with a casual look at the graphic, might help in summarizing multi-modal documents.[1]

1 Our colleagues are currently investigating how the findings from this work can be used in communicating the content of multimodal documents.

3. System Overview

Figure 2 provides an overview of the overall system architecture. The inputs to our system are an XML representation of a bar chart and the intended message of the chart; the former is the responsibility of a Visual Extraction System (Chester and Elzer 2005) and the latter is the responsibility of a Bayesian Inference System (Elzer, Carberry, and Zukerman 2011). Given these inputs, the Content Identification Module (**CIM**) first identifies the salient and important features of a graphic that are used to augment its inferred message in the summary. The propositions conveying the selected features and the inferred message of the graphic are then passed to the Text Structuring and Aggregation Module (**TSAM**). This module produces a partial ordering of the propositions according to the kind of information they convey, and aggregates them into sentence-sized units. The Sentence Ordering Module (**SOM**) then determines the final ordering of the sentence-sized units. Finally, the Sentence Generation Module (**SGM**) realizes these units in natural language, giving particular attention to generating referring expressions for graphical elements when appropriate. In the rest of this section, we briefly present the systems that provide input to our work and describe the corpus of bar charts used for developing and testing our system. The following sections then describe the modules implemented within our system in greater detail, starting from the Content Identification Module.

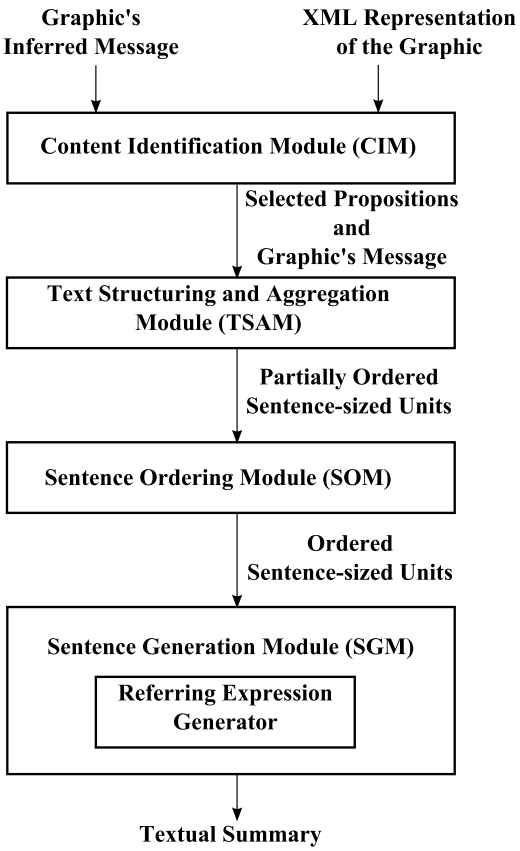


Figure 2
System architecture.

3.1 Visual Extraction System

The Visual Extraction System (Chester and Elzer 2005) analyzes a graphic image (visual image of a bar chart) and creates an XML representation specifying the components of the graphic, such as the height and color of each bar, any annotations on a bar, the caption of the graphic, and so forth. The current implementation handles vertical and horizontal bar charts that are clearly drawn with specific fonts and no overlapping characters. The charts can have a variety of textual components such as axis labels, caption, further descriptive text, text inside the graphic, and text below the graphic. The current system cannot handle 3D charts, charts where the bars are represented by icons, or charts containing texts at multiple angles, however.

3.2 Bayesian Inference System for Intention Recognition

The Bayesian Inference System (Elzer, Carberry, and Zukerman 2011) treats an information graphic as a form of language with a communicative intention, and reasons about the communicative signals present in the graphic to recognize its intended message. The system is currently limited to simple bar charts and takes as input the XML representation of the chart produced by the Visual Extraction System described previously.

Three kinds of communicative signals that appear in bar charts are extracted from a graphic and utilized by the system. The first kind of signal is the relative effort required for various perceptual and cognitive tasks. The system adopts the AutoBrief (Kerpedjiev and Roth 2000) hypothesis that the graphic designer chooses the best design to facilitate the perceptual and cognitive tasks that a viewer will need to perform on the graphic. Thus, the relative effort for different perceptual tasks serves as a communicative signal about what message the graphic designer intended to convey (Elzer et al. 2006). The second and third types of communicative signals used in the system are salience and the presence of certain verbs and adjectives in the caption that suggest a particular message category. The presence of any of these three kinds of communicative signals are entered into a Bayesian network as evidence. The top level of the network captures one of the 12 message categories that have been identified as the kinds of messages that can be conveyed by a bar chart, such as conveying a change in trend (**Changing Trend**) or conveying the bar with the highest value (**Maximum Bar**). The system produces as output the hypothesized intended message of a bar chart as one of these 12 message categories, along with the instantiated parameters of the message category, in the form of a logical representation such as Maximum Bar(first bar) for the graphic in Figure 1 and Increasing Trend(first bar, last bar) for the graphic in Figure 3a.

3.3 Corpus of Graphics

We collected 82 groups of graphics along with their articles from 11 different magazines (such as Newsweek and Business Week) and newspapers. These groups of graphics varied in their structural organization: 60% consisted solely of a simple bar chart (e.g., the graphic in Figure 1 on Page 2) and 40% were composite graphics (e.g., the graphic in Figure 8a in Section 7.1.1) consisting of at least one simple bar chart along with other bar charts or other kinds of graphics (e.g., stacked bar charts or line graphs). We selected at least one simple bar chart from each group and our corpus contained a total of 107 bar charts. The Bayesian Inference System had an overall success rate of 79.1% in recognizing the correct intended message for the bar charts in our corpus using leave-one-out cross-validation (Elzer, Carberry, and Zukerman 2011).

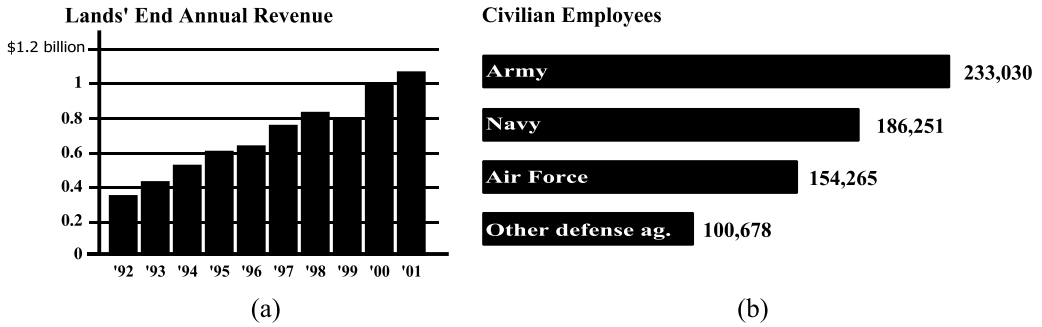


Figure 3
(a) Graphic conveying an increasing trend. (b) Graphic conveying the ranking of all bars.

In the work described in this article, we only used the bar charts whose intended message was correctly recognized by the Bayesian Inference System and associated each chart with the inferred message category. Here, our intent is to describe a generation approach that works through a novel problem from beginning to end by handling a multitude of generation issues. Thus, using bar charts with the perfect intention is reasonably appropriate within the scope of the present work. For each bar chart, we also used the XML representation that was utilized by the Bayesian Inference System. Slightly less than half of the selected bar charts were kept for testing the system performance (which we refer to as the **test corpus**), and the remaining graphs were used for developing the system (which we refer to as the **development corpus**). Because the number of graphics in the development corpus was quite limited, we constructed a number of bar charts[2] in order to examine the effects of individual salient features observed in the graphics from the development corpus. These graphs, most of which were obtained by modifying original graphics, enabled us to increase the number of graphics in the development corpus and to explore the system behavior in various new cases.

4. Content Identification Module (CIM)

Our ultimate goal is to generate a brief and coherent summary of a graphic. Identifying and realizing the high-level informational content of a graphic is not an easy task, however. First, a graphic depicts a large amount of information and therefore it would be impractical to attempt to provide all of this information textually to a user. Second, a graphic is chosen as the communication medium because a reader can get information from it at many different levels. A casual look at the graphic is likely to convey the intended message of the graphic and its salient features. At the same time, a reader could spend much more time examining the graphic to further investigate something of interest or something they noticed during their casual glance.

In order to address the task of identifying the content of a summary, we extend to simple bar charts the insights obtained from an informal experiment where human participants were asked to write a brief summary of a series of line graphs with the same high-level intention (McCoy et al. 2001). The most important insight gained from

2 The graphics that we constructed were not used in any of the evaluation experiments with human participants described throughout this article.

this study is that the intended message of a graphic was conveyed in all summaries no matter what the visual features of the graphic were. It was observed that the participants augmented the intended message with salient features of the graphic (e.g., if a line graph is displaying an increasing trend and the variance in that trend is large, then the variance is salient) and that what was found salient depended on the graphic's intended message. Because the participants generated similar summaries for a particular graphic, we hypothesize that they perceived the same salient features for that graphic. Although the set of features that might be salient is the same for different graphics sharing the same underlying intention, the differences observed between the summaries generated for different graphics with the same intention can be explained by whether or not the features are salient in those graphics. The fact that the summaries did not include all information that could be extracted from the graphic (such as the value of every point in a line graph) but only visually salient features, correlates with Grice's Maxim of Quantity (1975) which states that one's discourse contribution should be as informative as necessary for the purposes of the current exchange but not more so.

To extend these observations to constructing brief summaries of bar charts, we hypothesize that (1) the intended message of the bar chart should form the core of its textual summary and (2) the most significant and salient features of the bar chart, which are related to its intended message, should be identified and included in that summary. The inferred intended message of a bar chart serves as a starting point for our content identification approach. In the rest of this section, we first describe a series of experiments that we conducted to identify what constitutes the salient features of a given bar chart and in which circumstances these features should be included in its textual summary. We then present the content identification rules that were constructed to automatically select appropriate content for the summary of a bar chart.

4.1 Experiments

We conducted a set of formal experiments to find patterns between the intended message of a graphic, salient visual features of the displayed data, and the propositions selected for inclusion in a brief summary. We identified the set of all propositions (**PROPALL**) that capture information that we envisioned someone might determine by looking at a bar chart. This set included a wide variety of pieces of information present in a bar chart and contained propositions common to all bar charts as well as propositions which were applicable only to some of the message categories. The following is a subset of the identified propositions. In this example, Propositions 1–4 are common to all bar charts; in contrast, Propositions 5–8 are only present when the bar chart is intended to convey a trend:

- The labels of all bars (**Proposition 1**)
- The value of a bar (**Proposition 2**)
- The percentage difference between the values of two bars (**Proposition 3**)
- The average of all bar values (**Proposition 4**)
- The range of the bar values in the trend (**Proposition 5**)
- The overall percentage change in the trend (**Proposition 6**)

- The change observed at a time period (**Proposition 7**)
- The difference between the largest and the smallest changes observed in the trend (**Proposition 8**)

Some propositions, which we refer to as open propositions, require instantiation (such as Propositions 2, 3, and 7 given here) and the information that they convey varies according to their instantiations.[3] In addition, the instantiation of an open proposition may duplicate another proposition. For example, if the Proposition 3 is instantiated with the first and the last bars of the trend, then the information conveyed by that proposition is exactly the same as Proposition 6.

To keep the size of the experiment reasonable, we selected 8 message categories from among the 12 categories that could be recognized by the Bayesian Inference System; these categories were the ones most frequently observed in our corpus and could be used as a model for the remaining message categories. These categories were Increasing Trend, Decreasing Trend, Changing Trend, Contrast Point with Trend, Maximum Bar, Rank Bar, Rank All, and Relative Difference. In the experiments, we did not use the categories Minimum Bar (which can be modeled via Maximum Bar), Relative Difference with Degree (which can be modeled via Relative Difference), Stable Trend (which was not observed in the corpus), and Present Data (which is the default category selected when the system cannot infer an intended message for the graphic).

For each message category, we selected two to three original graphics from the development corpus, where the graphics with the same intended message presented different visual features. For example, we selected two graphics conveying that a particular bar has the highest value among the bars listed, but only in one of these graphics was the value of the maximum bar significantly larger than the values of the other bars (such as the graphic in Figure 1). In total, 21 graphics were used in the experiments and these graphics covered all selected intended message categories. Because the number of propositions applicable to each message category was quite large, 10–12 propositions were presented for each graphic. Each graphic was presented to at least four participants. Overall, the experiments covered all selected intended message categories and all identified propositions.

Twenty participants, who were unaware of our system, participated in the experiments. The participants were graduate students or recent Ph.D. graduates from a variety of departments at the University of Delaware. Each experiment started with a brief description of the task, where the participants were told to assume that in each case the graphic was part of an article that the user is reading and that the most important information depicted in the graphic should be conveyed in its summary. They were also told that they would be given an information graphic along with a sentence conveying the intended message of the graphic and a set of propositions, and would be asked to classify these additional propositions into one of three classes according to how important they felt it was to include that proposition in the textual summary:[4] (1) **Essential**: This proposition should be included in the brief textual summary, (2) **Possible**: This proposition could be included in the brief textual summary but it's not essential, and (3) **Not Important**: This proposition should not be included in the brief textual summary.

3 We used open propositions in order to keep PROPALL within a manageable size.

4 The participants were also asked to instantiate the open propositions that they classified as Essential or Possible.

4.2 Analysis

To analyze the experiment's results, we first assigned a numeric score to each class indicating the level of importance assigned by the participants: Essential = 3, Possible = 1, Not-important = 0. We then calculated an "importance level" (**IL**) for each proposition with respect to a particular graphic, where the importance level estimates how important it is for that proposition to be included in the graphic's summary. The importance level of a proposition was computed by summing the numeric scores associated with the classes assigned by the participants. For example, if three participants classified a proposition as Essential and two participants as Possible, the importance level of that proposition in the graphic was $(3 \times 3) + (2 \times 1) = 11$. In cases where a proposition (Prop A) and an instantiated open proposition which conveyed the same information were classified by a participant into different classes for the same graphic, the classification of the proposition that came earlier in the presentation was used in computing the importance level of Prop A.

Given these computed scores, we needed to identify which propositions to consider further for inclusion in a summary. Because there was a divergence between the sets of propositions that were classified as essential by different participants, we decided to capture the general tendency of the participants. For this purpose, we defined **majority importance level** as a ranking criteria, which is the importance level that would be obtained if half of the participants classify a proposition as essential. For example, the majority importance level would be $(6 \times 3)/2 = 9$ if there were six participants. We classified a proposition as a highly rated proposition if its importance level was equal to or above the majority importance level.[5] The propositions that were classified as highly rated for the graphics with a message category formed the set of highly rated propositions that should be considered for inclusion for that message category.

We had to ensure that the propositions presented to the participants (PROPALL) actually covered all information that is important enough to include in the summary of a bar chart. Thus, for each graphic, we also asked participants if there was anything else they felt should be included in the brief summary of the graphic. We received only a few isolated suggestions such as a proposition conveying what type of a curve could fit the trend. Moreover, these suggestions were not common among the participants, and nothing was mentioned by more than one participant (indeed most did not make any suggestions). Thus, we concluded that these suggestions were not appropriate for the textual summary of a bar chart.

4.3 Content Identification Rules for Message Categories

Using the importance level scores, we needed to identify the subset of the highly rated propositions that should be included in the textual summary in addition to the graphic's intended message. For each message category, we examined the similarities and the differences between the sets of highly rated propositions identified for the graphics

5 The reason behind assigning particular scores (3,1,0) to the classes is to guarantee that a proposition will not be selected as a highly rated proposition if none of the participants thought that it was essential. Assume k participants classified a proposition (Prop A). The majority importance level of this proposition (MIL(Prop A)) is $\lceil (3 \times k)/2 \rceil$. A proposition is classified as highly rated if its importance level (IL(Prop A)) is equal to or greater than the majority importance level (IL(Prop A) $\geq$ MIL(Prop A)). If all of the participants classified the proposition as Possible, the IL(Prop A) is $1 \times k$, which is less than MIL(Prop A).

associated with that message category, related these differences to the visual features present in these graphics, and constructed a set of content identification rules for identifying propositions to be included in the summary of a graphic from that message category. If a proposition was marked as highly rated for all graphics in a particular message category, then its selection was not dependent on particular visual features present in these graphics. In such cases, our content identification rule simply states that the proposition should be included in the textual summary for every graphic whose inferred message falls into that message category. For the other propositions that are highly rated for only a subset of the graphics in a message category, we identified a feature that was present in the graphics where the proposition was marked as highly rated and was absent when it was not marked as highly rated, and our content identification rules use the presence of this feature in the graphic as a condition for the proposition to be included in the textual summary. In addition, we observed that a highly rated proposition for a message category might require inclusion of another proposition for realization purposes. For example, in the Rank All message category, the proposition indicating the rank of each bar was identified as highly rated and thus could be included in the textual summary. Because the rank of a bar cannot be conveyed without its label, we added the proposition indicating the label of each bar to the content identification rule containing the rank proposition, although this extra proposition was not explicitly selected by the participants for inclusion. Notice that these steps—identifying features that distinguish one subset of graphs from the other and identifying propositions that need to be included for realizing other propositions—make it difficult to use machine learning for this task. In our case the number of possible features that can be extracted from a graphic is very large and it is difficult to know which features from among those may be important/defining in advance. In addition, the number of graphics in our development corpus is too small to expect machine learning to be effective.

The following are glosses of two partial sets of representative content identification rules. The first set is applicable to a graphic conveying an increasing trend and the second set is applicable to a graphic conveying the rankings of all bars present in the graph:

- **Increasing Trend message category:**[6]

  1. If (message category **equals** 'increasing trend') then **include**(proposition conveying the rate of increase of the trend):
     Include the proposition conveying the rate of increase of the trend

  2. If (message category **equals** 'increasing trend') and **notsteady**[7](trend) then **include**(proposition conveying the period(s) with a decrease):[8]
     If the trend is not steady and has variability, then include the proposition indicating where the trend varies

6 The "notsteady" function returns true if its argument is not a steady trend; the "value" function returns the values of all members of its argument; the "greaterthan" function returns true if the left argument is greater than the right argument; the "withinrange" function returns true if all members of its left argument are within the range given by its right argument; the "average" function returns the average of the values of all members of its argument.

7 A trend is unsteady if there is at least one period with a decrease in contrast with the increasing trend.

8 The inclusion of propositions whose absence might lead the user to draw false conclusions is consistent with Joshi, Webber, and Weischedel's (1984) maxim, which states that a system should not only produce correct information but should also prevent the user from drawing false inferences.

3. If (message category **equals** 'increasing trend') and (**value**(last bar) **greaterthan** (3***value**(first bar))) then **include**(proposition conveying the overall percentage increase in the trend):
If the overall percentage increase in the trend is significantly large, then include the proposition conveying the percentage increase in the trend

- **Rank All message category**:

1. If (message category **equals** 'rank all') then **include**(propositions conveying the label and the value of the highest bar):
Include the propositions conveying the label and the value of the highest bar

2. If (message category **equals** 'rank all') and (**value**(all bars) **withinrange** ((0.7***average**(all bars)),(1.3***average**(all bars)))) then **include**(proposition indicating that the bar values vary slightly):
If the values of bars are close to each other, then include the proposition indicating that the bar values vary slightly

3. If (message category **equals** 'rank all') and (**not**(**value**(all bars) **withinrange** ((0.7***average**(all bars)),(1.3***average**(all bars))))) then **include**(propositions conveying the label and the value of the lowest bar):
If the values of bars are not close to each other, then include the propositions conveying the label and the value of the lowest bar

We defined the conditions of all content identification rules as a conjunction of one or more expressions where some expressions required us to determine threshold values to be used for comparison purposes. For example, we observed that the proposition conveying the overall percentage change in the trend was marked as highly rated only for graphics which depicted a significant change in the trend. We handled this situation for graphics with an increasing trend by defining the third content identification rule (shown earlier) where we needed to set the lowest threshold at which an overall increase observed in a trend can be accepted as significantly large. For setting such threshold values, we examined all graphs in the development corpus to which the corresponding content identification rule is applicable (i.e., the graphs associated with the message category for which the rule is defined) and used our intuitions about whether the proposition captured by the rule should be selected for inclusion in the summaries of these graphs. We set the threshold values using the results obtained from group discussions such that the final setting classified all of the original graphics the way the participants did in the experiments described in Section 4.1.

When the content identification rules constructed for the Increasing Trend message category are applied to the bar chart in Figure 3a, the following pieces of information are selected for inclusion in addition to the intended message of the graphic:

- The rate of increase of the trend, which is slight
- The small drop observed in the year 1999
- The overall percentage increase in the trend, which is 225%

When the content identification rules constructed for the Rank All message category are applied to the bar chart in Figure 3b, the following pieces of information are selected for inclusion in addition to the intended message of the graphic:

- The label and the value of the highest bar, which is Army with 233,030
- The label and the value of the lowest bar, which is Other defense agencies with 100,678
- The label and the ranking of each bar:[9] Army is the highest, Navy is the second highest, Air Force is the third highest, and Other defense agencies is the lowest

4.4 Evaluation of the Content Identification Module

We conducted a user study to assess the effectiveness of our content identification module in identifying the most important information that should be conveyed about a bar chart. More specifically, the study had three goals: (1) to determine whether the set of highly rated propositions that we identified for each message category contains all propositions that should be considered for inclusion in the summaries of graphics with that message category; (2) to determine how successful our content identification rules are in selecting highly rated propositions for inclusion in the summary; and (3) to determine whether the information conveyed by the highly rated propositions is misleading or not.

Nineteen students majoring in different disciplines (such as Computer Science and Materials Science and Engineering) at the University of Delaware were participants in the study. These students neither participated in the earlier study described in Section 4.1 nor were aware of our system. Twelve graphics from the test corpus (described in Section 3.3) whose intended message was correctly identified by the Bayesian Inference System were used in the experiments. Once the intended message was recognized, the corresponding content identification rules were executed in order to determine the content of the graphic's summary. Prior to the experiment, all participants were told that they would be given a summary and that it should include the most important information that they thought should be conveyed about the graphic. Each participant was presented with three graphics from among the selected graphics such that each graphic was viewed by at least four participants. For each graphic, the participants were first given the summary of the graphic generated by our approach and then shown the graphic. The participants were then asked to specify if there was anything omitted that they thought was important and therefore should be included in the summary. In addition, the participants were asked to specify whether or not they were surprised or felt that the summary was misleading (i.e., whether the bar chart was similar to what they expected to see after reading its summary). Note that our summaries with relatively few propositions are quite short. Thus our evaluation focused on determining whether anything of importance was missing from the summary or whether the summary was misleading. In the experiments, we did not ask the participants to rate

9 This piece of information is selected by a rule defined for the Rank All message category not shown in the bulleted list on the previous page.

the content of summaries on a numeric scale in order to restrict them to evaluating only the selected content as opposed to its presentation (i.e., the organization and realization of the summary).

Feedback that we received from the participants was very promising. In most of the cases (43 out of 57 cases), the participants were satisfied with the content that our approach selected for the presented bar charts. There were a number of suggestions for what should be added to the summaries in addition to what had already been conveyed, and in a couple of these cases, we observed that a highly rated proposition which was not selected by the corresponding content identification rule was contrarily suggested by the participants. There was no consensus in these suggestions, however, as none was made by more than two participants. Some of the participants (3 out of 19) even commented that we provided more information than they could easily get from just looking at the graphic. In addition, a few participants (2 out of 19) commented that, in some graphics, they didn't agree with the degree (e.g., moderate or steep) assessed by our approach for differences between bar values (e.g., the rate of change of the trend), and therefore they thought the summary was misleading. Because there wasn't any common consensus among the participants, we didn't address this very subjective issue. Overall, we conclude that the sets of highly rated propositions that we identified contain the most important information that should be considered for inclusion in the summaries of bar charts and that our system effectively selects highly rated propositions for inclusion when appropriate.

5. Text Structuring and Aggregation Module (TSAM)

A coherent text has an underlying structure where the informational content is presented in some particular order. Good text structure and information ordering have proven to enhance the text's quality by improving user comprehension. For example, Barzilay, Elhadad, and McKeown (2002) showed that the ordering has a significant impact on the overall quality of the summaries generated in the MULTIGEN system. Although previous research highlights a variety of structuring techniques, there are three prominent approaches that we looked to for guidance: top–down planning, application of schemata, and bottom–up planning.

In top–down planning (Hovy 1988, 1993; Moore and Paris 1993), the assumption is that a discourse is coherent if the hearer can recognize the communicative role of each of its segments and the relation between these segments (generally mapped from the set of relations defined in rhetorical structure theory (RST; Mann and Thompson 1987). The discourse is usually represented as a tree-like structure and the planner constructs a text plan by applying plan operators starting from the initial goal.

In the TEXT system (McKeown 1985), a collection of naturally occurring texts were analyzed to identify certain discourse patterns for different discourse goals, and these patterns were represented as schemas which are defined in terms of rhetorical predicates. The schemas both specify what should be included in the generated texts and how they should be ordered given a discourse goal. Lester and Porter (1997) used explanation design packages, schema-like structures with procedural constructs (for example, the inclusion of a proposition can be constrained by a condition), in the KNIGHT system, which is designed to generate explanations from a large-scale biology knowledge base. Paris (1988) applied the idea of schemata in the TAILOR system to tailor object descriptions according to the user's level of knowledge about the domain.

Marcu (1998) argued that text coherence can be achieved by satisfying local constraints on ordering and clustering of semantic units to be realized. He developed a constraint satisfaction based approach to select the best plan that can be constructed from a given set of textual units and RST relations between them, and showed that such bottom–up planning overcomes the major weakness of top–down approaches by guaranteeing that all semantic units are subsumed by the resulting text plan. The ILEX system (O'Donnell et al. 2001), which generates descriptions for exhibits in a museum gallery, utilizes a similar bottom–up planning approach (Mellish et al. 1998) where the best rhetorical structure tree over the semantic units is used as the text structure.

Because our content identification rules identify a set of propositions to be conveyed, it appears that a bottom–up approach that ensures that all propositions will be included is in order. At the same time, it is important that our generated text adheres to an overall discourse organization such as is provided by the top–down approaches. Because of the nature of the propositions (the kinds of rhetorical relations that can exist between propositions in a descriptive domain are arguably limited [O'Donnell et al. 2001]), however, a structure such as RST is not helpful here. Thus, the top–down planning approach does not appear to fit. Although something akin to a schema might work, it is not clear that our individual propositions fit into the kind of patterns used in the schema-based approach. Instead we use what can be considered a combination of a schema and a bottom–up approach to structure the discourse. In particular, we use the notion of global focus (Grosz and Sidner 1986) and group together propositions according to the kind of information they convey about the graphic. We define three proposition classes (**message-related, specific,** and **computational**) to classify the propositions selected for inclusion in a textual summary. The message-related class contains propositions that convey the intended message of the graphic. The specific class contains the propositions that focus on specific pieces of information in the graphic, such as the proposition conveying the period with an exceptional drop in a graphic with an increasing trend or the proposition conveying the period with a change which is significantly larger than the changes observed in other periods in a graphic with a trend. Lastly, propositions in the computational class capture computations or abstractions over the whole graphic, such as the proposition conveying the rate of increase in a graphic with an increasing trend or the proposition conveying the overall percentage change in the trend. In our system, all propositions within a class will be delivered as a block. But we must decide how to order these blocks with respect to each other. In order to emphasize the intended message of the graphic (the most important piece of the summary), we hypothesize that the message-related propositions should be presented first. We also hypothesize that it is appropriate to close the textual summary by bringing the whole graphic back into the user's focus of attention (Grosz and Sidner 1986) (via the propositions in the computational class). Thus, we define an ordering of the proposition classes (creating a partial ordering over the propositions) and present first the message-related propositions, then the specific propositions, and finally the computational propositions. Section 6 will address the issue of ordering the propositions within these three classes.

5.1 Representing Summary Content

First we needed to have a representation of content that would provide us with the most flexibility in structuring and realizing content. For this we used a set of basic propositions. These were minimal information units that could be combined to form

the intended message and all of the propositions identified in our content identification rules. This representation scheme increases the number of aggregation and realization possibilities that could be explored by the system, which is described in the next subsection. We defined two kinds of knowledge-base predicates to represent the basic propositions:

(1) **Relative Knowledge Base:** These predicates are used to represent the basic propositions which introduce graphical elements or express relations between the graphical elements.

(2) **Attributive Knowledge Base:** These predicates are used to represent the basic propositions which present an attribute or a characteristic of a graphical element.

Each predicate contains at least two arguments and we refer to the first argument as the **main entity** and the others as the **secondary entities**. The main entity of each predicate is a graphical element and the secondary entities are either a string constant or a graphical element. Some of the graphical elements that we used in this work are as follows:

- **graphic:** "the graphic itself"
- **trend:** "the trend observed in the graphic"
- **descriptor:** "a referring expression that represents what is being measured in the graphic"[10]
- **bar(x):** "a particular bar in the graphic"
  $1 <= x <= n$ *where* $n =$ *number of bars in the graph*
- **all bars:** "all bars depicted in the graphic" $bset = \{bar(x) \mid 1 <= x <= n\}$
- **period(x,y):** "a period depicted in the graphic" $1 \leq x < n$ *and* $1 < y \leq n$
- **change(x,y):** " the change between the values of any two bars"
  $1 \leq x < n$ *and* $1 < y \leq n$
- **all changes:** "changes between all pairs of bars of the graphic"
  $cset = \{change(x, y) \mid 1 \leq x < n, 1 < y \leq n\}$
- **trend period:** "the period over which the trend is observed"
- **graph period:** "the period which is depicted by the graphic"
- **trend change:** "the overall change observed in the trend"

Table 1 presents sample instantiations of a subset of the predicates that we defined for this work along with a possible realization for each instantiation. Although the number of arguments in Relative Knowledge Base predicates (predicates 1 to 15) varies,

10 How that referring expression is extracted from the text associated with the graphic is described in detail in Section 7.1. For example, the descriptor identified by our system for the graphic in Figure 4 is *the dollar value of net profit*.

Table 1
Sample instantiations and possible realizations of a subset of our predicates.

1	**shows(graphic,trend)** *The graphic shows a trend*
2	**focuses(graphic,bar(3))** *The graphic is about the third bar*
3	**covers(graphic, graph period)** *The graphic covers the graph period*
4	**exists(trend,descriptor)** *The trend is in the descriptor*
5	**has(trend,trend period)**[11] *The trend is over the trend period*
6	**starts(trend period,"2001")** *The trend period starts at the year 2001*
7	**ends(trend period,"2010")** *The trend period ends at the year 2010*
8	**ranges(descriptor,"from","20 percent",trend period)** *The descriptor ranges from 20 percent over the trend period*
9	**hasextreme(descriptor,"largest",change(3,4),period(3,4))** *The descriptor shows the largest change between the third and the fourth bars*
10	**averages(descriptor,all bars,"55.4 billion dollars")** *The descriptor for all bars averages to 55.4 billion dollars*
11	**comprises(descriptor,trend change,trend period)** *The descriptor comprises a trend change over the trend period*
12	**occurs(change(2,3),period(2,3))** *A change occurs between the second and the third bars*
13	**hasdifference(change(1,5),bar(1),bar(5),descriptor)** *A difference is observed between the descriptor of the first bar and that of the fifth bar*
14	**observed(all changes,"every",interval,trend period)** *Changes are observed every interval over the trend period*
15	**presents(descriptor,bar(3),"12 percent")** *The descriptor for the third bar is 12 percent*
16	**hasattr(trend change,"type","increase")** *The trend change is an increase*
17	**hasattr(change(2,3),"degree","moderate")** *The change is of degree moderate*
18	**hasattr(change(2,3),"amount","70 dollars")** *The change amount is 70 dollars*
19	**hasattr(trend change,"percentage amount","22 percent")** *The trend change percentage amount is 22 percent*
20	**hasattr(all changes,"rate","slight")** *Changes are slight changes*

all Attributive Knowledge Base predicates (encoded as *hasattr*) consist of three arguments, where the first argument is the graphical element being described, the second is an attribute of the graphical element, and the third is the value of that attribute (predicates 16 to 20).[12]

11 Notice that the graphical element trend period in 5 is the main entity in 6 and 7. These all might be combined using the And operator to produce the realization *The trend starts at the year 2001 and ends at the year 2010*.

12 Because all Attributive Knowledge Base predicates have the same form, the amount and unit of a change are represented as a single string which is derived from the textual components of the graphic (such as *70 dollars* in Predicate 18).

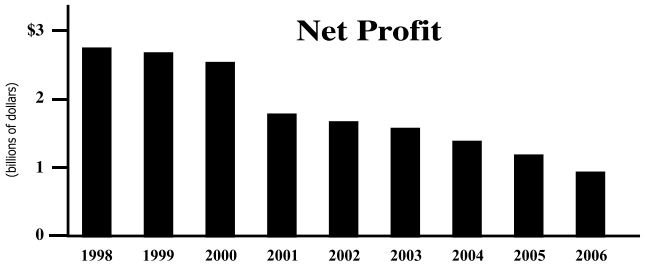


Figure 4
Graphic conveying a decreasing trend.

For example, consider how the propositions given in Section 4.1 can be represented with the predicates shown in Table 1. Some of those propositions require a single predicate. For example, the proposition conveying the value of a bar (Proposition 2) can be represented via the predicate "presents" (Predicate 15) and the proposition conveying the average of all bar values (Proposition 4) via the predicate "averages" (Predicate 10). On the other hand, some propositions require more than one predicate. For example, the proposition conveying the overall percentage change in the trend shown in Figure 4 (Proposition 6) can be represented via the predicates "comprises(descriptor,trend change,trend period)" (Predicate 11), "starts(trend period,"1998")" (Predicate 6), "ends(trend period,"2006")" (Predicate 7), "hasattr(trend change,"type","decrease")" (Predicate 16), and "hasattr(trend change,"percentage amount","65 percent")"(Predicate 19). The same set of predicates can be used to represent the overall amount of change in the trend by replacing the constant "percentage amount" with the string "amount" in Predicate 19.

As is shown by the possible realizations included in Table 1, each basic proposition can be realized as a single sentence. Although we determined a couple of different ways (i.e., simple sentences) of realizing each basic proposition, our current implementation always chooses a single realization (which we refer to as "the realization associated with the proposition") and the main entity is always realized in subject position.[13]

5.2 Aggregating Summary Content

The straightforward way of presenting the informational content of a summary is to convey each proposition as a single sentence while preserving the partial ordering of the proposition classes. The resultant text would not be very natural and coherent, however. Aggregation is the process of removing redundancies during the generation of a more concise and fluent text (Shaw 1998; Dalianis 1999). Aggregation (typically syntactic aggregation [Reiter and Dale 2000]) has received considerable attention from the NLG community (McKeown et al. 1997; O'Donnell et al. 2001; Barzilay and Lapata 2006), and has been applied in various existing generation systems such as the intelligent tutoring application developed by Di Eugenio et al. (2005). Our aggregation mechanism works to combine propositions into more complex structures. It takes

13 We leave it as a future work to explore how different realizations for a proposition, including ones where the main entity is not in subject position, can be utilized by our approach.

advantage of the two types of predicates (Relative Knowledge Base and Attributive Knowledge Base predicates) and the shared entities between predicates. In order to relate propositions and explore syntactically aggregating them, our mechanism treats each proposition as a single node tree which can be realized as a sentence and attempts to form more complex trees by combining individual trees via four kinds of operators in such a way so that the more complex tree (containing multiple propositions) can still be realized as a single sentence. The first operator (Attribute Operator) works only on propositions with an Attributive Knowledge Base predicate and essentially identifies opportunities to realize such a proposition as an adjective attached to a noun object in the realization of another proposition. The remaining three operators, which do not work on propositions with an Attributive Knowledge Base predicate, introduce new nodes corresponding to operational predicates (And, Same, and Which) with a single entity into the tree structures. Two of these operators (And Operator and Which Operator) work on trees rooted by a proposition with a Relative Knowledge Base or an And predicate. These operators look for opportunities for VP conjunction and relative clauses, respectively. The third operator (Same Operator) works on trees rooted by a proposition with a Relative Knowledge Base predicate and identifies opportunities for NP conjunction. Although each predicate is associated with a unique realization in the current implementation, none of these operators depend on how the corresponding predicates or the entities in those predicates are realized.

Having defined the operators we next had to turn to the problem of determining how these operators should be applied (e.g., which combinations are preferred). The operators we defined are similar to the clause-combining operations used by the SPoT sentence planner (Walker, Rambow, and Rogati 2002; Stent, Prasad, and Walker 2004; Walker et al. 2007) in the travel planner system AMELIA. In AMELIA, for each of the 100 different input text plans, a set of possible sentence plans (up to 20 plans) were generated by randomly selecting which operations to apply according to assumed preferences for operations. These possible sentence plans were then rated by two judges and the collected ratings were used to train the SPoT planner. Although we greatly drew from the work on SPoT as we developed our aggregation method, we chose not to follow their learning methodology. In the SPoT system, some of the features were domain- and task-dependent and thus porting to a new domain would require retraining. In addition, the judgments of the two raters were collected in isolation and it is unclear how these would translate to the task situation the texts were intended for. Although this methodology was innovative and necessary for SPoT because of the large number of possible text plans, we chose to select the best text plan on the basis of theoretically informed complexity features balancing sentence complexity and number of sentences. Because our text plans are significantly more constrained, it is possible to enumerate each of them and choose the one that best fits our rating criteria.[14] This has the added benefit of better understanding the complexity features by evaluating the resulting text. In addition, our method would be open to both upgrading the selection criteria and adding further aggregation operators without requiring retraining.[15]

14 Although in our implementation we do enumerate all plans before the rating criteria are applied to select the best one, it is in principle possible to generate the text plans in an order that would allow maximizing the scoring functions without first enumerating all possibilities. This is left for future work.

15 Such modifications and additions need to be empirically evaluated with empirical data, however.

Operator: Attribute Operator
Gloss: This operator attaches a single node tree that consists solely of a proposition with an Attributive Knowledge Base predicate, as a direct subchild of a node N with a Relative Knowledge Base predicate in another tree, if the main entity of the Attributive Knowledge Base predicate is an entity (main or secondary) for the proposition at node N.
Input: T1 and T2
Constraints:
1. (pred(T1-root)==“hasattr”)
2. ((pred(T2-node)!=“hasattr”) ∧ (pred(T2-node)!=“And”) ∧
(pred(T2-node)!=“Which”))
3. ((main ent(T1-root)==main ent(T2-node)) ∨
(main ent(T1-root)==secondary ent(T2-node)))
Output: A modified T2 such that
1. left child(T2-node)←T1
Glossary:
1. Tx-root: the root node of tree Tx
2. Tx-node: any node in tree Tx (including Tx-root)
3. pred(Tx-node): the predicate at Tx-node
4. left/right child(Tx-node): the leftmost/rightmost child of Tx-node
5. main/secondary ent(Tx-node): the main/secondary entity of the proposition at Tx-node
6. !=: not equal, ==: equal, !: not, ←: assign

Operator: And Operator
Gloss: This operator combines two trees if the propositions at their root share the same main entity. A proposition containing an And predicate with the same main entity forms the root of the new tree and the trees that are combined form the immediate descendents of this root.
Input: T1 and T2
Constraints:
1. ((pred(T1-root)!=“hasattr”) ∧ (pred(T2-root)!=“hasattr”))
2. !((pred(T1-root)==“And”) ∧ (pred(T2-root)==“And”))
3. (main ent(T1-root)==main ent(T2-root))
Output: a new tree T3 where the root node has two immediate children such that
1. pred(T3-root)←“And” ∧ main ent(T3-root)←main ent(T1-root)
2. left child(T3-root)←T1 ∧ right child(T3-root)←T2

Operator: Which Operator
Gloss: This operator attaches a tree (Tree A) as a descendent of a node N in another tree (Tree B) via a Which predicate, if the main entity of the proposition at the root of Tree A is a secondary entity for the proposition at node N of the other tree (Tree B). That particular entity forms the main entity of the Which predicate. Thus, Tree A will be an immediate child of the node with the Which predicate and the node with the Which predicate will be an immediate child of node N in Tree B.
Input: T1 and T2
Constraints:
1. ((pred(T1-root)!=“hasattr”)∧(pred(T2-node)!=“hasattr”))
2. (main ent(T1-root)==secondary ent(T2-node))
Output: A modified T2 via the addition of a new node (Node x) with a single immediate child such that
1. pred(Node x)← “Which” ∧ main ent(Node x)←main ent(T1-root)
2. right child(T2-node)← Node x ∧ left child(Node x)←T1

Operator: Same Operator
Gloss: This operator combines two trees if the propositions at their root contain the same predicate but the main entities of these predicates are different. A proposition with a Same predicate forms the root of the new tree, and the trees that are combined form the descendents of this root. Since the descendents of the new tree have different main entities, the main entity of the Same predicate is some unique element not occurring elsewhere in the tree. For instance, in our implementation this element is obtained by appending a unique number, which isn't used in another Same predicate in the current forest, to the term *random* (such as *random0*).
Input: T1 and T2
Constraints:
1. ((pred(T1-root)!="hasattr")∧(pred(T2-root)!="hasattr") ∧
(pred(T1-root)!="And") ∧ (pred(T2-root)!="And") ∧
(pred(T1-root)!="Same") ∧ (pred(T2-root)!="Same"))
2. (pred(T1-root)==pred(T2-root))
3. (main ent(T1-root)!=main ent(T2-root))
Output: a new tree T3 where the root node has two immediate children such that
1. pred(T3-root)←"Same" ∧ main ent(T3-root)← a unique element
2. left child(T3-root)←T1 ∧ right child(T3-root)←T2

In our work, we thus developed a method that would choose a text plan on principled reasoning concerning the resulting text. In particular, we looked to balance sentence complexity and the number of sentences in the generated text. Moreover, whereas such a method was not applicable in the case of SPoT (because of the significantly larger set of operators with few constraints resulting in potential text plans too numerous to evaluate), our work differs in several aspects that make it reasonable to generate all text plans and apply an evaluation metric. First, our system has a small number of aggregation operators and all operators cannot be applied to all kinds of predicates (e.g., the Attribute Operator cannot be applied to the Relative Knowledge Base predicates). Second, the number of possible sets of basic propositions that our system needs to organize is significantly lower than the number of possible text plans that the SPoT planner needs to consider. Finally, although it is not practical in SPoT to list all possible sentence plans that might be generated for a particular text plan (since the possibilities are too great), generating all possible combinations of propositions in a proposition class (such as message related class) is practical in our work. This is due to the fact that the number of basic propositions in each class is fairly small (e.g., usually between 5 to 15 propositions) and that the nature of the operators and constraints that we put on their application enable us to prune the space of possible combinations. Some of these constraints are: (1) The And Operator produces only one complex tree from a pair of trees and it cannot combine two trees if both trees have a proposition with an And predicate at their roots (thus we limit the number of conjuncts in a conjoined sentence to three at most[16]), and (2) the Attribute Operator produces only one complex tree in cases where a single node tree (Tree A) can be attached as a direct subchild of more than one node in another tree (Tree B); the parent of Tree A is the first such node found by preorder traversal of Tree B.

Our implementation first generates all possible text plans for the propositions within each class (message-related, specific, and computational). Each text plan is represented as a forest where each tree in the forest represents a sentence. Initially, each proposition class is treated as a forest consisting of all single node trees in that class (initial candidate forest), and the operators are applied to that forest in order to produce new candidate forests for the proposition class. Anytime two trees in a

16 We set this limit in order to avoid sentences that are too complex to comprehend.

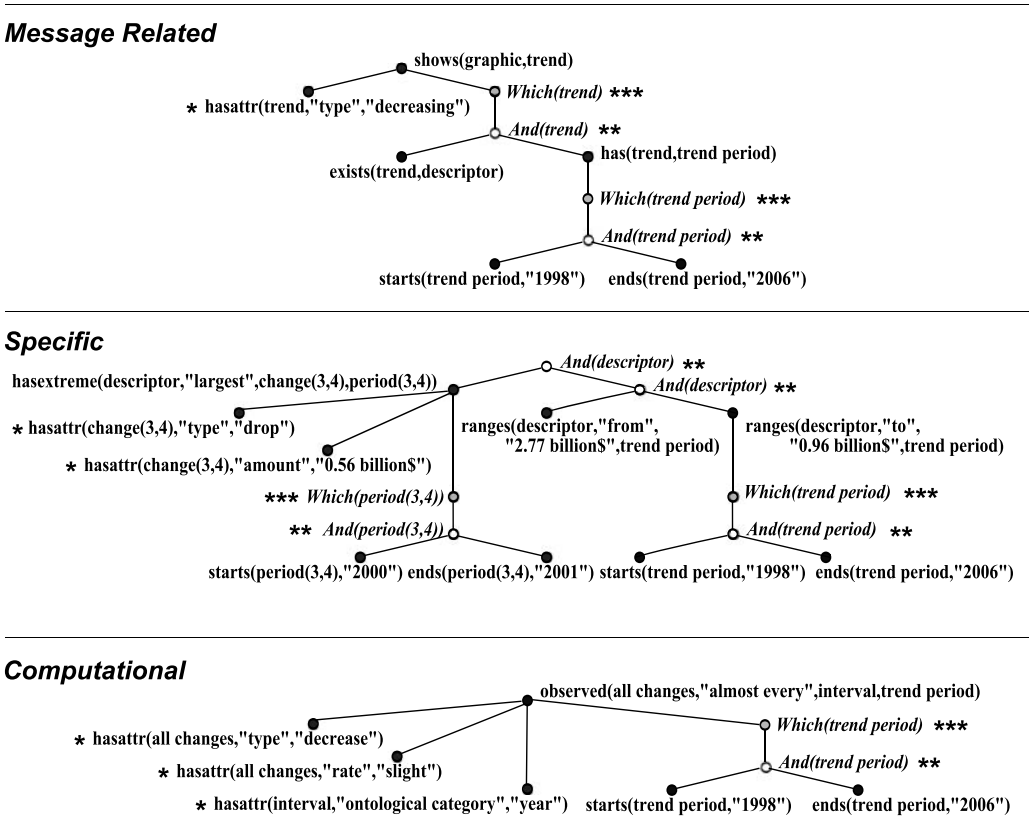


Figure 5
A candidate forest for each proposition class.

candidate forest are combined via an operator, a new candidate forest is produced; the new candidate forest is added to the existing set of candidate forests, thereby increasing the number of candidate forests. Within each class, our approach first applies the And Operator to all possible pairs of trees in the initial candidate forest, which produces new candidate forests. The Same Operator is then applied to all possible pairs of trees in each candidate forest. Similarly, the Which Operator and the Attribute Operator are applied to trees in the candidate forests produced earlier. The result of this aggregation is a number of candidate forests with one or more trees (each using different aggregation) for each of the proposition classes. For example, Figure 5 shows one candidate forest that can be constructed for each proposition class by applying these operators to the propositions selected for the graphic in Figure 4, where each forest resulting from the aggregation consists of a single tree.[17] In this example, the Attributive Knowledge Base predicates **(*)** are attached to their parents by the Attribute Operator, the nodes containing And predicates **(**)** are produced by the And Operator, and the Which predicates **(***)** are produced by the Which Operator.

17 The nodes represented with black circles correspond to the individual predicates. These individual predicates within each class form the single node trees upon which the operators work.

5.3 Evaluating a Text Structure

Different combinations of operators produce different candidate forests in each proposition class and consequently lead to different realized text with a different complexity of sentences. The set of candidate forests for each proposition class must be evaluated to determine which one is best. Our objective is to find a forest that would produce text which stands at a midpoint between two extremes: a text where each proposition is realized as a single sentence and a text where groups of propositions are realized with sentences that are too complex. Our evaluation metric to identify the best forest leverages different considerations to balance these extremes. The first two criteria are concerned with the number and syntactic complexity of sentences that will be used to realize a forest. The third criteria takes into consideration how hard it is to comprehend the relative clauses embedded in these sentences. The insights that we use in selecting the best forest (e.g., balancing semantic importance, overall text structure, aggregation, and readibility due to sentence complexity) represent our novel contributions to the text structuring and aggregation literature. The theory that underlies our evaluation metric (i.e., what it is we are balancing in the generation) is widely applicable to other data-to-text generation domains because it uses general principles from the literature and has the potential to be improved.

5.3.1 Sentence Complexity. Each tree (single node or complex) in a forest represents a set of propositions that can be realized as a single sentence. Our aggregation rules enable us to combine these simple sentences into more complex syntactic structures. In the literature, different measures to assess syntactic complexity of written text and spoken language samples have been proposed, with different considerations such as the right branching nature of English (Yngve 1960) and dependency distance between lexemes (Lin 1996). We apply the revised D-level sentence complexity scale (Covington et al. 2006) as the basis of our syntactic complexity measure. The D-level scale measures the complexity of a sentence according to its syntactic structure and the sequence in which children acquire the ability to use different types of syntactic structures. The sentence types with the lowest score are those that children acquire first and therefore are the simplest types. Eight levels are defined in the study, some of which are simple sentences, coordinated structures (conjoined phrases or sentences), non-finite clause in adjunct positions, and sentences with more than one level of relative clause embedding.

Among the eight levels defined in that study, the levels of interest in our work are simple sentences, conjoined sentences, sentences with a relative clause modifying the object of the main verb, non-finite clauses in adjunct positions, and sentences with more than one level of embedding. However, the definition of sentence types at each level is too general. For example, the sentences *There is a trend* and *There is a trend in the dollar value of net profit over the period from the year 1998 to the year 2006* are both classified as simple sentences with the lowest complexity score under the D-level classification. We argue that although these sentences have a lower complexity than the sentences with higher D-level scores, their complexities are not the same. We make a finer distinction between sentence types defined in the D-level classification and use the complexity levels shown in Table 2. For example, according to our classification, a simple sentence with more than one adjunct or preposition has a higher complexity than a simple sentence without an adjunct. We preserve the ordering of the complexity levels in the D-level classification. For example, in our classification,

Table 2
Our syntactic complexity levels.

Complexity	Syntactic Form
Level 0	Simple sentence with up to one prepositional phrase or adjunct
Level 1	Simple sentence with more than one prepositional phrase or adjunct
Level 2	Conjoined sentence (two simple sentences—Level 0 or 1)
Level 3	Conjoined sentence (more than two simple sentences—Level 0 or 1)
Level 4	Sentence with one level of embedding (relative clause that is modifying object of main verb)
Level 5	Non-finite clause in adjunct positions
Level 6	Sentence with more than one level of embedding

Levels 0 and 1 correspond to the class of simple sentences in the D-level classification and have a lower complexity than Levels 2 and 3, which correspond to the class of coordinated structures with a higher complexity than simple sentences in the D-level classification.

Each basic proposition in our system can be realized as a simple sentence containing at most one prepositional phrase or adjunct.[18] Each single node tree with a Relative or Attributive Knowledge Base predicate at its root has the lowest syntactic complexity (Level 0) in this classification.

The most straightforward way of realizing a more complex tree would be conjoining the realizations of subtrees rooted by a proposition with an And or a Same predicate, embedding the realization of a subtree rooted by a proposition with a Which predicate as a relative clause, and realizing a subtree that consists solely of a proposition with an Attributive Knowledge Base predicate as an adjective or a prepositional phrase. For example, the tree rooted by shows(graphic,trend) in Figure 5 can be realized as *The graphic shows a decreasing trend, which is in the dollar value of net profit and is over the period, which starts at the year 1998 and ends at the year 2006*. The resultant text is fairly complicated, however, and a more sophisticated realization would likely lead to a lower syntactic complexity score. We defined a number of And predicate and Which predicate complexity estimators to look for realization opportunities in a complex tree structure so that a syntactic complexity score which is lower than what the most straightforward realization would produce can be assigned to that tree. These estimators compute the syntactic complexity of a complex tree by examining the associated realizations of all aggregated propositions in that tree in a bottom–up fashion. Because the complex trees that are rooted by a proposition with a Same predicate would always be realized as a conjoined sentence (Level 2), we did not define complexity estimators for this kind of predicate.

The **And predicate complexity estimators** check whether or not the realizations of two subtrees rooted by a proposition with an And predicate can be combined into a simple sentence (Level 1), or a conjoined sentence which consists of two independent sentences (Level 2) if one of the subtrees is rooted by a proposition with an And predicate. For example, the And predicate estimators can successfully identify the

18 In the current implementation, there is a single realization associated with each basic proposition with the main entity in subject position.

following realization opportunities (based on the representations of the sentences as propositions):

- *The period starts at the year 1998.* AND *The period ends at the year 2006.* can be combined into:
  The period is from the year 1998 to the year 2006. (Level 1)
- *The trend is in the dollar value of net profit.* AND *The trend is over the period from the year 1998 to the year 2006.* can be combined into:
  The trend is in the dollar value of net profit over the period from the year 1998 to the year 2006. (Level 1)
- *The dollar value of net profit ranges from 2.77 billion dollars over the period.* AND *The dollar value of net profit ranges to 0.96 billion dollars over the period. AND The dollar value of net profit shows the largest drop of about 0.56 billion dollars between the year 2000 and the year 2001.* can be combined into:[19]
  The dollar value of net profit ranges from 2.77 to 0.96 billion dollars over the period and shows the largest drop of about 0.56 billion dollars between the year 2000 and the year 2001. (Level 2)

The **Which predicate complexity estimators** check whether a tree rooted by a proposition with a Which predicate can be realized as a simple adjunct or a prepositional phrase attached to the modified entity rather than a more complex relative clause (which could increase the complexity level). For example, the Which predicate estimators can successfully identify the following realization opportunities (based on the representations of the sentences as propositions):

- *The trend is over the period.* WHICH *The period is from the year 1998 to the year 2006.* can be realized as:
  The trend is over the period from the year 1998 to the year 2006. (Level 1)
- *The graphic shows a trend.* WHICH *The trend is in the dollar value of net profit over the period from the year 1998 to the year 2006.* can be realized as:
  The graphic shows a trend in the dollar value of net profit over the period from the year 1998 to the year 2006. (Level 1)

In our generation approach, multiple realizations for each proposition can be incorporated by defining new complexity estimators in addition to the estimators that are used in the current implementation. Defining such estimators, which will not change the task complexity or the underlying methodology, would add to the generalizability of our approach.

19 In this case, the single node trees that correspond to the propositions conveying the range of the trend form the immediate descendents of a tree rooted by a proposition with an And predicate, and that tree with the And predicate at its root and the tree corresponding to the proposition conveying the largest drop constitute the immediate descendents of a tree rooted by another proposition with an And predicate.

5.3.2 Relative Clause Embedding. In cases where a tree rooted by a proposition with a Which predicate cannot be realized as a simple adjunct or a prepositional phrase, it will be realized by a relative clause. In the D-level classification, the complexity of a sentence with an embedded clause is determined according to the grammatical role (subject or object) of the entity that is modified by that clause, not the syntactic complexity or position (center-embedded or right-branching) of the clause in the sentence. For instance, a sentence with a complex center-embedded relative clause modifying an object receives the same syntactic complexity score as a sentence with a simple right-branching relative clause modifying an object. As argued in the literature, however, center-embedded relative clauses are more difficult to comprehend than corresponding right-branching clauses (Johnson 1998; Kidd and Bavin 2002). To capture this, our evaluation metric for identifying the best structure penalizes Which predicates that will be realized as a relative clause based on the clause's syntactic complexity and position in the sentence (which we refer to as "comprehension complexity of a relative clause"). For example, the following sentences receive different scores by our evaluation metric with respect to clause embedding; the first one with a right-branching clause (simpler) has a lower score than the second sentence with a center-embedded clause (more complex):

- *The graphic shows a decreasing trend over the period from the year 1998 to the year 2006 in the dollar value of net profit, which is 2.7 billion dollars in the year 1999.*
- *The graphic shows a decreasing trend in the dollar value of net profit, which is 2.7 billion dollars in the year 1999, over the period from the year 1998 to the year 2006.*

The embedded clause (Level 0) in the first of the following sentences has a lower syntactic score than the clause (Level 2) embedded in the second sentence. Because our evaluation metric takes into consideration both the syntactic complexity of an embedded clause and its position in the sentence, the first sentence receives a lower score than the second sentence.

- *The graphic shows a decreasing trend in the dollar value of net profit, which is 2.7 billion dollars in the year 1999, over the period from the year 1998 to the year 2006.*
- *The graphic shows a decreasing trend in the dollar value of net profit, which is 2.7 billion dollars in the year 1999 and is 2.58 billion dollars in the year 2000, over the period from the year 1998 to the year 2006.*

5.3.3 Evaluation Metric. Our evaluation metric takes three criteria into account: the number of sentences that will be used to realize a forest, the syntactic complexities of these sentences, and the comprehension complexities of the embedded relative clauses. Our metric evaluates the overall score of a candidate forest by summing the normalized scores that the forest receives with respect to each criteria. The score of a forest (e.g., Forest A) is calculated as follows:

$$score(\mathbf{A}) = nm_1(sentence(\mathbf{A})) + nm_2(complexity(\mathbf{A})) + nm_3(clause(\mathbf{A})) \quad (1)$$

where:[20]

sentence(A): stands for the number of sentences that will be used to realize forest A and equals the number of trees in that forest.

complexity(A): stands for the overall syntactic complexity of forest A and equals the sum of the complexities of sentences that will be used to realize that forest.

clause(A): stands for the overall comprehension complexity of all relative clauses in sentences used for realizing forest A and equals the sum of the comprehension complexities of all clauses. The comprehension complexity of a relative clause equals the product of its syntactic complexity and its position in the sentence, which is equal to 2 if it is a center-embedded clause and is equal to 1 if it is a right-branching clause.

Consider, for example, the forest shown in Figure 6. Because the forest contains a single tree, it receives a score of 1 for the sentence criteria. The syntactic complexity score of the sentence that will be used to realize that tree is computed in a bottom–up fashion as follows. The lowest syntactic complexity score (Level 0) is assigned to all leaf nodes, and all inner nodes that only have single node trees with an Attributive Knowledge Base predicate as descendents (as shown in Figure 6). Each of the remaining inner nodes is then assessed with a syntactic complexity score once the complexity scores for all of its descendents are computed (i.e., once the best realization possibility with the lowest syntactic complexity for each descendent tree is determined). If an inner node contains a proposition with an And predicate, its syntactic complexity score is computed via the And predicate complexity estimators. Similarly, the Which predicate complexity estimators are used to compute the syntactic complexity scores for all inner nodes with a Which predicate. The syntactic complexity score for the parent node of a tree rooted by a proposition with a Which predicate is computed based on whether or not that tree will be realized as a relative clause (as indicated by the complexity score of the root node of that tree). The forest shown in Figure 6 receives a score of 4 for the complexity criteria, which is equal to the syntactic complexity score assigned to the parent node of the tree. In Figure 6, only the tree rooted by Node 4 will be realized as a relative clause. Because that relative clause, which receives a syntactic complexity score of 2, will be realized as a center-embedded clause, the forest shown in Figure 6 receives a score of 4 for the clause criteria.

In the current implementation, once the scores with respect to a criteria are computed for each candidate forest, these scores (e.g., **sentence(A)**) are normalized with respect to the maximum score (e.g., **max(sentence(all forests))**) by dividing each score by the maximum of the computed scores. For instance, $\mathrm{nm}_1(\mathrm{sentence}(\mathbf{A}))$ is the normalized score that the forest A receives with respect to the sentence criteria and is equal to sentence(A)/max(sentence(all forests)). Thus, the normalized score that a forest receives for each criteria is always between 0 and 1 and therefore each criteria has an equal impact on the overall score of a forest.[21] The normalized scores obtained for a candidate forest are then summed to obtain the overall score for that forest. For example, assume that three candidate forests, the first of which is shown in Figure 6,

20 The terms $\mathbf{nm}_1$, $\mathbf{nm}_2$, and $\mathbf{nm}_3$ stand for the normalized score of a given criteria.

21 The simplifying assumption of assigning equal weights to each criteria would be better optimized with machine learning, as discussed in detail in Section 9.

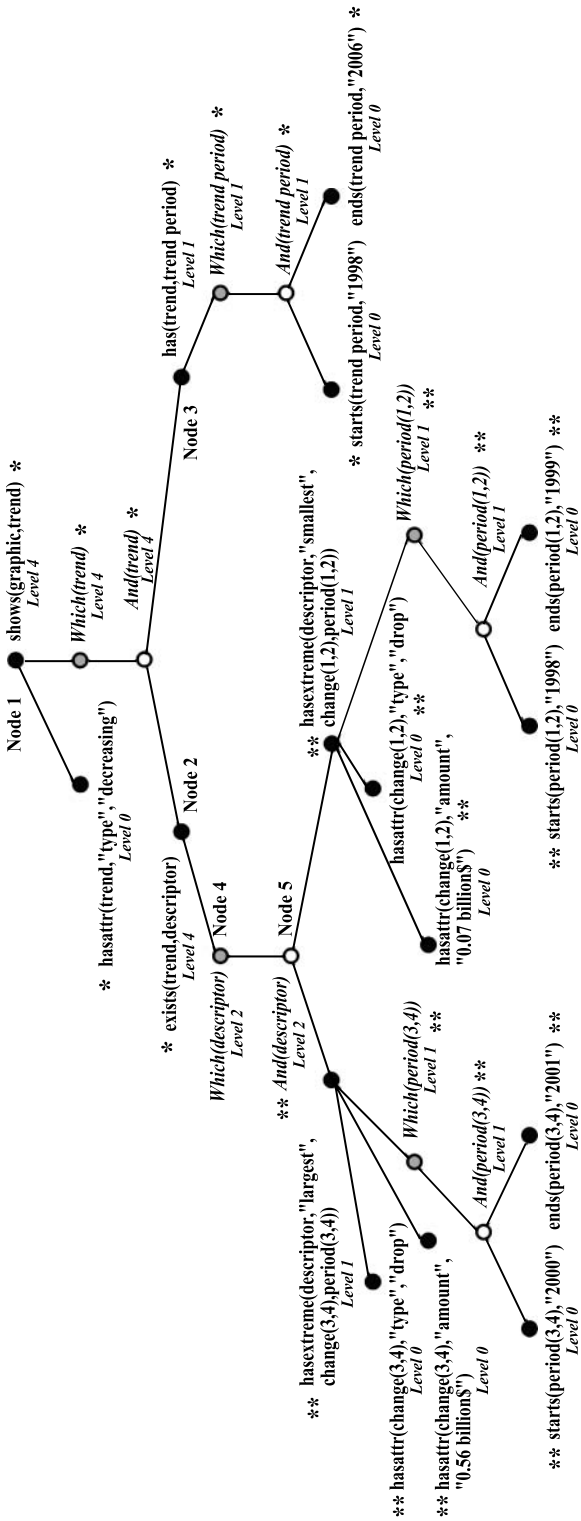


Figure 6
A forest containing a single tree.

Table 3
Overall evaluation scores.

	Sentence	Complexity	Clause	Overall Score
First Forest	1(0.5)	4(1)	4(1)	2.5
Second Forest	1(0.5)	4(1)	2(0.5)	2
Third Forest	2(1)	3(0.75)	0(0)	1.75

are constructed from a set of propositions. One possible way of realizing the forest in Figure 6 would be *The graphic shows a decreasing trend in the dollar value of net profit, which shows the largest drop of about 0.56 billion dollars between the year 2000 and the year 2001, and shows the smallest drop of nearly 0.07 billion dollars between the year 1998 and the year 1999, over the period from the year 1998 to the year 2006.* Suppose that the second forest is similar to Figure 6 except that the children (Node 2 and Node 3) of the And(trend) node are swapped.[22] Suppose also that the third forest is similar to Figure 6 except that the tree is decomposed into two trees, which are rooted by Node 1 and Node 5, respectively. The first tree rooted by Node 1 consists of the nodes marked with **(*)** and the second tree rooted by Node 5 consists of the nodes marked with **(**)**. Table 3 shows the actual and the normalized scores (shown in parentheses) for each forest with respect to each criteria, and the overall score assigned by our evaluation metric.

The number of sentences (1) and the overall sentence complexity (Level 4) are the same for the first and second forests. The third forest has more sentences (2) but lower overall sentence complexity (Level 3) than the other two forests. The first forest has a center-embedded relative clause and receives a score of 4 for the **clause** criteria: the product of the complexity of the relative clause (2) and its position (2). On the other hand, the second forest has a right-branching relative clause and receives a score of 2 for the same criteria: the product of the complexity of the relative clause (2) and its position (1). The third forest doesn't have an embedded clause and receives a score of 0 for the clause criteria.

5.4 Identifying the Best Text Structure

Our approach selects the forest which receives the lowest evaluation score as the best forest that can be obtained from a set of input propositions. For example, according to the scores shown in Table 3, the third forest, which could be realized as *The graphic shows a decreasing trend in the dollar value of net profit over the period from the year 1998 to the year 2006. The dollar value of net profit shows the largest drop of about 0.56 billion dollars between the year 2000 and the year 2001 and shows the smallest drop of nearly 0.07 billion dollars between the year 1998 and the year 1999.*, would be selected as the best among the three forests. The initial overall text structure of a brief summary consists of the best forests identified for the message related, specific, and computational classes.

As a final step, we check whether we can improve the evaluation of the overall structure of the summary by moving trees (i.e., trees rooted by a Level-0 node such as

22 This swapping would cause the relative clause rooted by Node 4 to be a right-branching clause.

And(descriptor) in Figure 5, Specific) or subtrees (i.e., trees rooted by a Level-1 node with an And or a Relative Knowledge Base predicate such as hasextreme (descriptor,"largest",change(3,4),period(3,4)) in Figure 5, Specific) between the best forests for the three proposition classes. For example, the best forest for the specific class might contain a tree that conveys information about an entity introduced by a proposition in the message related class. Moving this tree to the message related class and using an operator to combine it with the tree introducing the entity might improve the evaluation of the overall structure of the summary. For example, for the graphic in Figure 4, this movement would allow our system to evaluate a structure where the tree rooted by the specific proposition And(descriptor) (shown in Figure 5, Specific) is attached as a descendent of the tree rooted by the message related proposition exists(trend,descriptor) (shown in Figure 5, Message Related) via a Which Operator. We explore all such possible movements between best forests for the proposition classes (if any) and determine the best overall text structure of the summary. To be consistent with the motivation behind the initial groupings of the propositions, we do not allow movements out of the message related class or any movement that will empty the computational class.

5.5 Evaluation of the Text Structuring and Aggregation Module

Our text structuring and aggregation approach consists of several different components, all of which contribute to the quality of the generated text. Our study focused on whether or not our decisions with respect to these components contributed to the perceived quality of the resultant summary: the organization and ordering (**O**) of the content (partial ordering of the propositions within classes and classification of the propositions), the aggregation (**A**) of the information into more complex tree structures (candidate forests constructed via operators), and the metric (**E**) used to evaluate candidate forests that represent different possible aggregations of the informational content.

We conducted an experiment with 15 participants (university students and graduate students) who were presented with six different summaries of twelve graphics from the test corpus (described in Section 3.3). The participants neither participated in earlier studies (described in Sections 4.1 and 4.4) nor were involved in this work. All presented summaries were automatically produced by our generation approach. The participants were not told how the presented summaries were produced (i.e., human-generated or computer-generated), however. We focused on graphics with an increasing or a decreasing trend, since these message categories exhibit the greatest variety of possible summaries. For each of the graphics, the participants were given a set of summaries in random order and asked to rank them in terms of their quality in conveying the content. The summaries varied according to the test parameters as follows:[23]

- **S_O+A+E+:** A summary that uses the ordering rules, the aggregation rules, and receives the lowest (best) overall score by the evaluation metric. This is the summary selected as best by the TSAM Module.

23 Although eight different summaries are logically possible with three different variables, we limited the number to four (the second and the third in which exactly one of the components was turned off and the fourth where all components were turned off) in order to keep the experiment within a manageable size.

Table 4
Ranking of summary types.

Summary Type	Best	2^{nd}	3^{rd}	4^{th}
S_O+A+E+	**65.6%**	26.6%	6.7%	1.1%
S_O+A+E-	16.7%	**32.2%**	33.3%	17.8%
S_O-A+E+	16.7%	30%	**40%**	13.3%
S_O-A-E-	1%	11.2%	20%	**67.8%**

- **S_O+A+E-:** A summary that uses the ordering and aggregation rules, but *does not* receive the lowest overall score by the evaluation metric. This is the summary that received the second lowest score.
- **S_O-A+E+:** A summary where the propositions are randomly ordered, but aggregation takes place, and it receives the lowest (best) overall score by the evaluation metric.
- **S_O-A-E-:** A summary consisting of single sentences that are randomly ordered.

Table 4 presents the results of the experiment. It is particularly noteworthy that the summary selected as the best by the Text Structuring and Aggregation Module was most often (65.6% of the time) rated as the best summary and overwhelmingly (92.2% of the time) rated as one of the top two summaries. The table shows that omitting the evaluation metric (S_O+A+E-) or omitting ordering of propositions (S_O-A+E+) results in summaries that are substantially less preferred by the participants. Overall, the results shown in Table 4 validate our ordering, aggregation, and evaluation methodology.

6. Sentence Ordering Module (SOM)

With the use of different kinds of operators and an evaluation metric, our system determines the partial ordering and the structure of sentences that will be used to realize the selected content but doesn't impose ordering constraints (final ordering) on the sentences within each proposition class. To decide in which sequence the sentences should be conveyed, we take advantage of the fact that each proposition has a defined main entity, which will be realized in the subject position of the sentence that will be used to realize the proposition. Identifying the subject of the realized sentences in advance allows us to use centering theory (Grosz, Weinstein, and Joshi 1995) to generate a text that is most coherent according to this theory.[24] The theory outlines the principles of local text coherence in terms of the way the discourse entities are introduced and discussed, and the transitions between successive utterances in terms of the entities in the hearer's center of attention. Although some fundamental concepts of the theory, such as the ranking of entities in an utterance, aren't explicitly specified, various researchers have applied centering theory to language generation (Kibble and Power 2004; Karamanis et al. 2009) with different interpretations. In our work, each sentence is regarded as an

24 If this assumption is relaxed, then centering theory would not be appropriate for an ordering component. In that case, a focusing theory such as McCoy and Cheng (1991), or Suri and McCoy (1994) could be used to order the sentences to be realized.

utterance. Following Brennan, Friedman, and Pollard (1987) and Grosz, Weinstein, and Joshi (1995), we rank the entities with respect to their grammatical functions where the entity in subject position is the most salient entity. When ordering sentences, we take into account the preference order for centering transitions: *continue* is preferred over *retain*, which is preferred over *smooth shift*, which is in turn preferred over *rough shift*.

For all message categories, the number of sentences in a proposition class would be limited (less than five) even if all of the highly rated propositions identified for that message category are selected for inclusion. Thus, a straightforward "generate and test" strategy is appropriate for ordering sentences in our case. For each proposition class, all possible orderings of the sentences within that class are generated. We assign different numeric scores to each centering transition, where *continue* receives a score of 3, and *retain* and *smooth shift* receive scores of 2 and 1, respectively. The *rough shift* transitions are assessed a score of 0. For each candidate ordering, we sum the scores for the kinds of transitions observed between consecutive sentences. The ordering that receives the highest score is selected as the best ordering for that proposition class. First, the best ordering of the sentences in the message related proposition class is selected. The subject of the last sentence in that ordering is used as the backward-looking center of the previous utterance when determining the best ordering of the sentences in the specific proposition class. Similarly, the subject of the last sentence in the best ordering for the specific class is used as the backward-looking center when identifying the best ordering for the computational proposition class.

For graphics that depict a time period, we also utilize the time periods mentioned in each conjunct of a conjoined sentence in order to specify in which order these conjuncts will be conveyed in the realized text. If the time periods mentioned in each conjunct of a conjoined sentence are different, these conjuncts are ordered such that the time period in focus in the first conjunct subsumes or precedes the time period in focus in the second conjunct. Consider how individual sentences in the following compound sentences are ordered by our approach:

- *The dollar value of net profit ranges from 2.77 to 0.96 billion dollars over the period from the year 1998 to the year 2006 and shows the largest drop of about 0.56 billion dollars between the year 2000 and the year 2001.*
- *The dollar value of net profit shows the smallest drop of nearly 0.07 billion dollars between the year 1998 and the year 1999 and shows the largest drop of about 0.56 billion dollars between the year 2000 and the year 2001.*

Note that in the first conjoined sentence, the time period mentioned in the first conjunct (from 1998 to 2006) subsumes the time period mentioned in the second conjunct (between 2000 and 2001). On the other hand, in the second conjoined sentence, the time period mentioned in the first conjunct (between 1998 to 1999) precedes the time period mentioned in the second conjunct (between 2000 and 2001).

7. Sentence Generation Module (SGM)

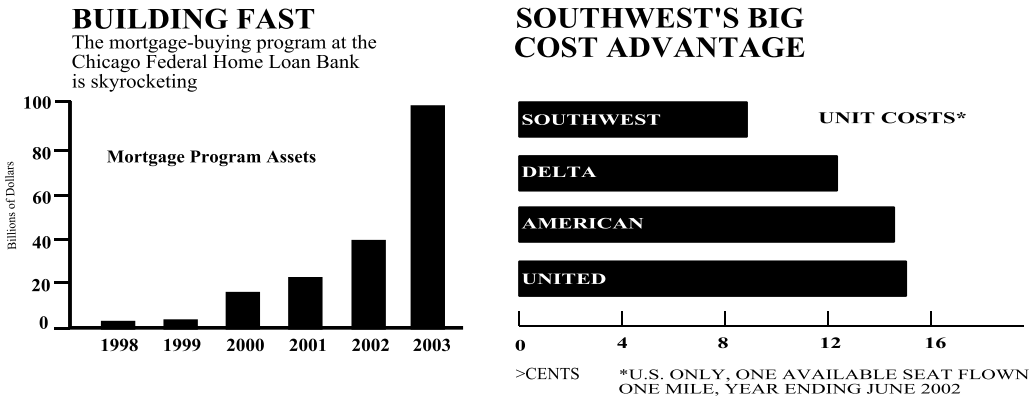
To realize the summaries in natural language, we use the FUF/SURGE surface realizer (Elhadad and Robin 1999), which offers the richest knowledge of English syntax and widest coverage among the publicly available realizers such as REALPRO (Lavoie and Rambow 1997). The realization of the sentence-sized units requires referring expressions

for certain graphical elements, however. Our system handles three different issues with respect to referring expression generation:

- Generating a referring expression for the dependent axis. Information graphics often do not label the dependent axis with a full descriptor of what is being measured (which we call the **measurement axis descriptor**). In such a situation, a referring expression for this element must be extracted from the text of the graphic. For example, to realize its summary, a measurement axis descriptor (e.g., *the dollar value of Chicago Federal Home Loan Bank's mortgage program assets*) must be generated for the graphic in Figure 7a, whose dependent axis is not explicitly labeled with the descriptor.
- Generating a referring expression in order to refer to the bars on the independent axis (e.g., *the countries* for the graphic in Figure 1). Such an expression must be inferred from the bar labels or extracted from the text of the graphic. This referring expression is often used in the summaries of graphics in some message categories (e.g., Maximum Bar) that require comparing a bar with others (e.g., distinguishing the bar with the maximum value from all other bars).
- It was shown that people prefer less informative descriptions for subsequent mentions of an entity (Krahmer and Theune 2002). In order to generate more natural summaries, the syntactic forms of the subsequent mentions of discourse entities should be constructed in a way that helps text coherence.

7.1 Measurement Axis Descriptor

Generation of referring expressions (noun phrases) is one of the key problems explored within the natural language generation literature. There is a growing body of research in this area that, given a knowledge base of entities and their properties, deals with



(a) (b)

Figure 7
(a) Graphic from *Business Week*. (b) Graphic from *Business Week*.

determining the set of properties that would single out the target entity (Dale and Reiter 1995; Krahmer, Van Erk, and Verleg 2003). More recently, the generation of referring expressions has been proposed as a postprocessing technique to deal with the lack of text coherence in extractive multidocument summarization (Belz et al. 2008). Nenkova and McKeown (2003) developed a method to improve the coherence of a multidocument summary of newswire texts by regenerating referring expressions for the first and subsequent mentions of people's names where the expressions are extracted from the text of the input documents according to a set of rewrite rules. The task that we face is similar to this recent body of research in that contextually appropriate referring expressions for certain graphical elements should be extracted from the text of the graphic. At the same time, our task is more complex in some respects. First, it is often the case that the required referring expression isn't explicitly given as a single unit and thus must be constructed by extracting and combining pieces of information from the text of the graphic. Second, in some cases where the dependent axis is explicitly labelled with a descriptor, it still needs to be augmented. We undertook a corpus study in order to identify how a measurement axis descriptor could be generated from the text of a graphic; the results of the analysis form the basis for the heuristics and augmentation rules we developed for generating the measurement axis descriptor for a graphic. In Demir, Carberry, and Elzer (2009), we outlined this problem as generating a graphical element required for realizing the intended message of a graphic and thoroughly described the technical details of our approach. Here, however, we treat this particular aspect as a novel text-to-text generation methodology which is combined with other data-to-text approaches in a complete NLG system. Thus, our focus in this section is to highlight a new way of using the text associated with images which has been earlier exploited by various NLP tasks such as indexing and retrieval of images (Pastra, Saggion, and Wilks 2003).

7.1.1 Corpus Analysis. Graphic designers generally use text within and around a graphic to present information related to the graphic. We started our analysis by examining how texts are distributed around each group of graphics. We observed that graphics (individual or composite) contain a set of component texts that are visually distinguished from one another by blank lines, by different fonts/styles, or by different directions and positions in the graphic. Although the number of component texts present in a graphic may vary, our analysis recognized an alignment or leveling of text contained in a graphic, which we refer to as "text levels."

We observed seven text levels which we refer to as Overall Caption, Overall Description, Caption, Description, Dependent Axis Label, Text In Graphic, and Text Under Graphic. Not every level appears in every graphic. Overall Caption and Overall Description apply to composite graphics that contain more than one individual graphic (the graphics might be of different kinds) and refer to the entire collection of graphics in the composite. In composite graphics, Overall Caption is the text that appears at the top of the overall group and serves as a caption for the whole set (such as *Tallying Up the Hits* in Figure 8a). In composite graphics, there is often another text component placed under the Overall Caption but distinguished from it by a line break or a change in font. This text component, if present, is also pertinent to all individual graphics in the composite graphic and elaborates on them. We refer to such text as the Overall Description (such as *Yahoo once relied entirely on banner ads. Now it's broadened its business mix* in Figure 8a). Caption and Description serve the same roles for an individual graphic. For example, the Caption for the bar chart in Figure 8a is *Active Users* and the Description is *Registered users in millions*. The Caption of Figure 8b

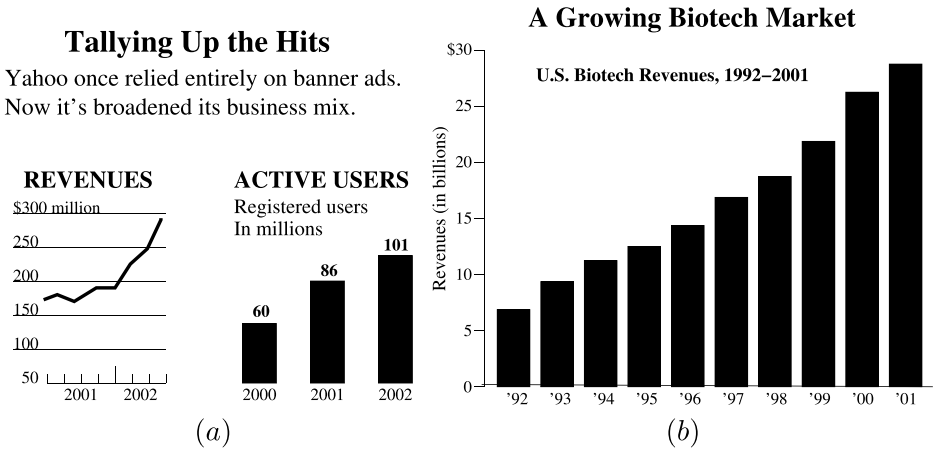


Figure 8
(a) A composite graph from *Newsweek*.[25] (b) Graphic from *Business Week*.

Table 5
Text levels in bar charts.

Text level	Frequency of occurrence
Overall Caption	31.8% (∼34/107)
Overall Description	17.8% (∼19/107)
Caption	99.0% (∼106/107)
Description	54.2% (∼58/107)
Text In Graphic	39.3% (∼42/107)
Dependent Axis Label	18.7% (∼20/107)
Text Under Graphic	7.5% (∼8/107)

is *A Growing Biotech Market* but this graphic does not have a Description. There may be a label on the dependent axis itself and we refer to it as Dependent Axis Label (such as *Revenues (in billions)* in Figure 8b). In addition to the text levels described so far which appear outside the borders of a graphic, we have observed that there is often a text component residing within the borders of a graphic which we refer to as Text In Graphic (such as *U.S. Biotech Revenues, 1992–2001* in Figure 8b). Finally, Text Under Graphic is the text under a graphic which usually starts with a marker symbol (such as *) and is essentially a footnote (such as *U.S. only, one available seat flown one mile, year ending June 2002* in Figure 7b). Each Text Under Graphic has a referrer elsewhere that ends with the same marker and that referrer could be at any text level of the graphic. A graphic might have more than one Text Under Graphic but each is differentiated with a different marker. For each of the 107 graphics in our corpus (described in Section 3.3), the Visual Extraction System extracts these text levels from the graphical image of the bar chart and inserts them into the graph's XML representation. Table 5 lists the various text levels, along with how often they appeared in the graphics in our corpus.

25 This figure displays two of the five individual graphs constituting the composite graphic that appeared in *Newsweek*.

Two annotators first analyzed each graphic in our corpus and determined a measurement axis descriptor for the graphic; the annotators used both the information residing within the text components of the graphic and the article, and commonsense knowledge. All ideal descriptors were noun phrases or *wh*-phrases[26] (such as *What's the most important issue affecting voters' vote?* on a graphic depicting survey results). After the descriptors were identified, we analyzed the graphics to see how the descriptors could be generated from the text components of the graphic. We observed that an acceptable measurement axis descriptor often cannot be extracted as a whole from a single text level and instead must be put together by extracting pieces from one or several different text levels; the pieces, though coming from a single level, might not be contiguous in that level and still need to be melded into a coherent whole. In some cases, the information is also retrieved from other graphics in the same composite or from the article's text. Our analysis has also led us to hypothesize that the ideal measurement axis descriptor can be viewed as consisting of a **core**—a basic noun or *wh*-phrase from one text level that is often augmented with text from another level (or in some cases, from text in the accompanying article or other graphs in the same composite) to be more descriptive and complete. For example, for the bar chart in Figure 8a, *registered users* is the core of the ideal measurement axis descriptor which is *Yahoo's registered users.* The core is found in the Description and the augmentation to the core is found in the Overall Description. When more than one text level is used, the text levels that contain pieces of the measurement axis descriptor also vary among the graphics. We observed that mostly text levels particular to a graphic (such as Text In Graphic and Description) contain the pieces of the descriptor as opposed to the levels containing shared information (such as Overall Description), and with the exception of Text Under Graphic, the ordering of text levels in Table 5 forms a hierarchy of textual components, with Overall Caption and Dependent Axis Label respectively at the top and bottom of the hierarchy, such that the core generally appears in the lowest text level present in the graphic. During the corpus analysis, we observed three ways in which a core extracted from one text level was augmented with text from another text level:

- **Expansion of the noun phrase:** The nouns in the core of the descriptor were replaced with a noun phrase at another text level which has the same noun as its head. The replaced noun phrase appeared in a text level higher in the precedence order than the text level at which the core appears. Consider, for example, Figure 8b. The core of the descriptor is *Revenues* (appearing in the Dependent Axis Label), which is reasonable enough to be the core, but the noun *Revenues* should be replaced with *U.S. Biotech Revenues* in order to be complete.
- **Specialization of the noun phrase:** The core was augmented with a proper noun which specialized the descriptor to a specific entity. Consider, for example, Figure 8a, which shows a composite graph where individual graphics present different attributes of the same particular entity (*Yahoo*). The ideal measurement axis descriptor of the bar chart (*Yahoo's registered users*) consists of the core *registered users* (appearing in the Description) augmented with the proper noun *Yahoo* that appears in the Overall Description.

26 Generally seen in graphics presenting the results of a survey.

- **Addition of detail:** Text Under Graphic typically serves as a footnote to give specialized detail about the graphic which is not as important as the information given in other text levels. If the Text Under Graphic began with a footnote marker, such as an asterisk, and the core was followed by the same marker, then Text Under Graphic added detail to the core. Consider, for example, Figure 7b, where *unit costs* is the core but the ideal measurement axis descriptor (*Unit costs, U.S. only, one available seat flown one mile, year ending June 2002*) also contains the information from the Text Under Graphic.

7.1.2 Methodology. First, preprocessing deletes the scale and unit indicators (phrases used to give the unit and/or a scale of the values presented in the graphic), and the ontological category of the bar labels (if explicitly marked by the preposition *by*) from the text levels. Next, heuristics are used to identify the core of the measurement axis descriptor by extracting a noun phrase or a *wh*-phrase from a text level of the graphic. Three kinds of augmentation rules, corresponding to the three kinds of augmentation observed in our corpus, are then applied to the core to produce the measurement axis descriptor. If none of the augmentation rules are applicable, then the core of the descriptor forms the measurement axis descriptor. Finally, if the measurement axis descriptor does not already contain the unit of measurement (such as *percent*), the phrase indicating the unit of measurement is appended to the front of the measurement axis descriptor.

For identifying the core of the measurement axis descriptor, we developed nine heuristics that are dependent on the parses of the text levels. Two of these heuristics are restricted to Dependent Axis Label and Text In Graphic, and the remaining heuristics are applicable to all other text levels. The application of the heuristics gives preference to text levels that are lower in the hierarchy and if a core is not identified at one text level, the applicable heuristics are applied, in order, to the next higher text level in the hierarchy. For example, suppose that the graphic contains only a Description and a Caption and thus the first two heuristics are not applicable. The next seven heuristics are first applied to the Description and then to the Caption. The following presents three representative heuristics where the first two heuristics are applicable only to Dependent Axis Label and Text In Graphic:[27]

- **Heuristic-1:** If the Dependent Axis Label consists of a single noun phrase that is not a scale or unit indicator, that noun phrase is the core of the measurement axis descriptor.
- **Heuristic-2:** If Text In Graphic consists solely of a noun phrase, then that noun phrase is the core; otherwise, if Text In Graphic is a sentence, the noun phrase that is the subject of the sentence is the core.
- **Heuristic-6:** If a fragment at the text level consists solely of a noun phrase, and the noun phrase is not a proper noun, that noun phrase is the core.

Once the core is identified, augmentation rules are applied to fill out the descriptor. For example, consider the graphic in Figure 8b where Heuristic-1 identifies *Revenues* in Dependent Axis Label as the core. Because the core and the Text In Graphic, *U.S. Biotech Revenues,* have the same head noun, the augmentation rule for expansion

27 All of the heuristics and augmentation rules can be found in Demir, Carberry, and Elzer (2009).

produces *U.S. Biotech Revenues* as the augmented core. After adding a phrase for the unit of measurement, the referring expression for the dependent axis becomes *The dollar value of U.S. Biotech Revenues.* As another example, consider the graphic in Figure 7b. Our work uses Heuristic-2 and the augmentation rule for adding detail. After adding a phrase representing the unit of measurement, the referring expression for the dependent axis becomes *The cent value of unit costs (U.S. only, one way available seat flown one mile, year ending June 2002)*. Finally, consider the graphic in Figure 8a, where Heuristic-6 identifies the noun phrase *registered users* as the core.[28] The augmentation rule for specialization finds that *Yahoo* is the only proper noun in the text levels and does not match a bar label, and forms *Yahoo's registered users.* After adding a phrase representing the unit of measurement, the referring expression for the dependent axis becomes *The number of Yahoo's registered users.*

In order to evaluate our approach, we constructed a distinct test corpus consisting of 205 randomly selected bar charts from 21 different newspapers and magazines; only six of these sources were also used to gather the corpus described in Section 3.3. For each graphic, we used our approach to generate the referring expression for the dependent axis. Finally, the resultant output and three baselines were evaluated by two evaluators (Demir, Carberry, and Elzer 2009). The evaluation results showed that our approach performs much better than any of the baselines for the 205 graphics in the corpus. The detailed analysis of the results also showed that our methodology is applicable to a wider range of sources in popular media.

7.2 Generating an Expression for Referring to All Bars

For some message categories (for example, Maximum Bar), the identification of the ontological category for the bar labels results in better natural language than merely using a generic expression; for example, compare the phrase *among the countries listed* with the phrase *among the entities listed* in producing natural language text for the message conveyed by the graphic in Figure 1. There are a number of different publicly available ontologies such as WordNet Fellbaum (1998) and OpenCyc (2011). In our work, we need a knowledge base that offers both the semantic relations between words and general commonsense knowledge. For example, WordNet could not identify *Jacques Chirac*, a former president of France, whereas OpenCyc ontology contains this information. Therefore, we use OpenCyc ontology version v0.7.8b to identify the ontological categories of bar labels in our work. Our implemented system currently finds the most specific category that is a common category for at least two of the bar labels and identifies it as the ontological category.

Grice's Maxim of Quantity (1975) states that one's discourse contribution should be as informative as necessary for the purposes of the exchange but not more so. If our system were to enumerate all entities involved in a comparison message, the realization of the inferred message might be lengthy and the enumeration of little utility to the user. Thus we set a cut-off C, such that if the number of entities involved in a Maximum Bar or Rank Bar message exceeds C, they are not enumerated but rather we use the generated referring expression for all bars. The cut-off value is currently set to 5 in our implemented system.

28 The preprocessing of this text level would remove *In millions* because it is a scale indicator.

7.3 Subsequent Mentions of Discourse Entities

In the current implementation of the system, the syntactic form of a subsequent mention of an entity is determined based on its salience status in the context. In particular, the backward-looking center of an utterance[29] is replaced with a less informative definite noun phrase after a *continue* or a *retain* transition because the backward-looking center remains the same in the latter utterance. In such cases, the definite noun phrase is constructed by adding the demonstrative *this* or *these* to the front of the head noun of the backward-looking center, such as *these revenues* for the phrase *U.S. biotech revenues.*

7.4 Example Summaries

For the graphic in Figure 1, our system generates the following textual summary: *The graphic shows that United States at 32,434 has the highest number of hacker attacks among the countries Brazil, Britain, Germany, Italy, and United States. United States has 5.93 times more attacks than the average of the other countries.*

For the graphic in Figure 3a, the following textual summary is generated: *The graphic shows an increasing trend in the dollar value of Lands' End annual revenue over the period from the year 1992 to the year 2001. The dollar value of Lands' End annual revenue shows an increase of nearly 225 percent. Except for a small drop in the year 1999, slight increases are observed almost every year.*

For the graphic in Figure 3b, our system generates the following summary: *The graphic compares the defense agencies army, navy, air force, and other defense agencies, which are sorted in descending order with respect to the number of civilian employees. The number of civilian employees is highest for army at 233,030 and is lowest for other defense agencies at 100,678.*[30]

For the graphic in Figure 4, the following textual summary is generated: *The graphic shows a decreasing trend in the dollar value of net profit over the period from the year 1998 to the year 2006. The dollar value of net profit ranges from 2.77 to 0.96 billion dollars over the period from the year 1998 to the year 2006 and shows the largest drop of about 0.56 billion dollars between the year 2000 and the year 2001. Slight decreases are observed almost every year.*

For the graphic in Figure 8b, our system generates the following summary: *The graphic shows an increasing trend in the dollar value of U.S. Biotech Revenues over the period from the year 1992 to the year 2001. Increasing slightly every year, the dollar value of U.S. Biotech Revenues shows an increase of nearly 265 percent and ranges from 7.87 to 28.52 billion dollars.*

For the graphic in Figure 7a, the following textual summary is generated: *In the year 2003, the graphic shows a much higher rise in the dollar value of Chicago Federal Home Loan Bank's mortgage program assets in contrast with the moderate increases over the period from the year 1998 to the year 2002. The dollar value of Chicago Federal Home Loan Bank's mortgage program assets reaches to 94.23 billion dollars in the year 2003. The dollar value of these assets in the year 2003 is nearly 49.1 times higher than that in the year 1998.*

For the graphic in Figure 9, the following textual summary is generated: *This graphic is about American Express. The graphic shows that American Express at 255 billion dollars is*

29 The backward-looking center of a current utterance is the most highly ranked entity of the previous utterance that is realized in the current utterance.

30 The reason for saying that the defense agencies are sorted in decreasing order is not to enable the reader to visualize the graphic but rather that it subsumes giving the ranking of each bar.

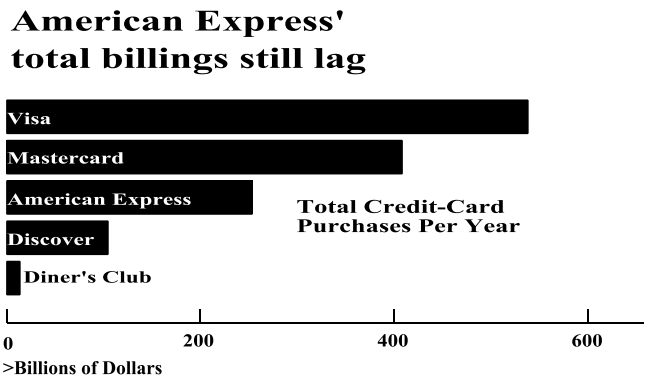


Figure 9
Graphic conveying the rank of a bar.

the third highest with respect to the dollar value of total credit-card purchases per year among the entities Visa, Mastercard, Discover, Diner's Club, and American Express.

8. Evaluation of the Effectiveness of the Textual Summaries

The earlier user studies (Sections 4.4 and 5.5) demonstrated the effectiveness of our generation methodology in identifying and presenting the high-level content of bar charts. The success of a generation system depends not only on the quality of the produced text, however, but also on whether the text achieves the impact that it is intended to make on readers (such as enabling readers to perform a task or changing their opinions in some context). We conducted an evaluation study to measure how adequate and effective the summaries generated by our system are for our purpose of providing the message and high-level knowledge that one would gain from viewing a graphic. Specifically, we were interested in (1) what amount of information is retained by a reader from reading the summary generated by our system, (2) whether someone reading the summary garners the most important knowledge conveyed by the graphic, and (3) whether the knowledge gleaned from the summary is consistent with the actual graphic.

In this study, we used four graphics from the test corpus (described in Section 3.3) with different intended messages. These graphics conveyed an increasing trend (i.e., Figure 3a), a decreasing trend, the rank of the maximum bar (i.e., Figure 1), and the rank of a bar (i.e., Figure 9) among all bars. In the first part of the study, each of the 18 participants (graduate students) was first presented with the summaries generated by our system for these graphics; the participants neither saw the original graphics (the graphical images of the bar charts) nor were aware of our system and how the summaries were generated. For each summary, the participants were asked to draw the bar chart being described in that summary. Although enabling a reader to redraw the graphic is not a goal of our work, comparing a reader's mental representation of the graphic with the actual graphic allows us to identify whether there are any inconsistencies between knowledge gleaned from the summary and the content of the actual graphic.

In the second part of the study, we asked three evaluators not involved in this research to evaluate the drawings that we collected from the participants. The evaluators were Ph.D. students from the University of Delaware (none of them were the authors of this work) and had an overall knowledge about our summarization approach (i.e., what is intended to be conveyed in the summaries of graphics). The evaluators were

first told that a set of participants were presented with brief summaries of bar charts and asked to draw the corresponding bar charts based on the information presented in those summaries. They were also told that each summary only conveyed what is identified by our system as the most important information that should be conveyed about the bar chart being described. The evaluators were presented with the graphical images of the four bar charts used in the study (none saw the summaries presented in the first part of the study) and the drawings collected from the participants, and then asked to rate each drawing using the following evaluation scores:

- **5:** The drawing is essentially the same as the original graphic
- **4:** The drawing captures all important information from the original graphic but requires some minor modifications
- **3:** The drawing captures most of the important information from the original graphic but is missing one significant piece of information
- **2:** The drawing reflects some information from the original graphic but requires major modifications
- **1:** The drawing fails to reflect the original graphic

The average score that a drawing received from the evaluators ranged from **3** to **5**. The evaluators viewed the drawings drawn for the graphics with a trend more favorably and assigned a score of 4 or more in most cases. For each graphic, we computed the average score given by the evaluators to all drawings constructed from the summary of that graphic (i.e., the average of the three scores given to each of the 18 drawings drawn from the same summary). The graphics conveying an increasing (Figure 3a) and a decreasing trend received a score of **4.22** and **4.63**, respectively. The evaluators gave an average score of **3.53** and **4.07** to the graphics which conveyed the rank of the maximum bar (Figure 1) and the rank of a bar among all bars (Figure 9). Because we do not present all features of a bar chart in its summary (such as all bar values), obtaining an average score of less than 5 for all bar charts is not surprising.

We also asked the evaluators to specify a reason (i.e., what is missing or should be changed) for the cases where they assigned a score of less than 4. Once we analyzed their feedback for the drawings with a trend, we observed that missing values on the dependent axis (i.e., tick mark labels) and missing measurement axis descriptors (although given in the summaries) were the main reasons. We argue that presenting tick mark labels is more appropriate for summaries that describe scientific graphics (such as the summaries generated by the iGRAPH-Lite system [Ferres et al. 2007]) in contrast to the summaries that we generate for conveying the high-level content of a graphic. The evaluators indicated incorrect rankings of some bars as the reason for giving lower scores to the drawings that present the rank of the maximum bar or the rank of a bar among all bars; this is due to the fact that our summaries did not convey the rankings and the values of all bars in those cases. Because the intention of the corresponding graphics is to convey the rank of a single bar (not all bars), we argue that our summaries facilitate the readers to get the main purpose of these graphics. Overall, this study demonstrated that our summarization approach is effective in conveying the high-level content of bar charts so as to enable readers to correctly understand the main point of the graphic. We are planning to conduct future studies to explore the effectiveness of our approach further, however. One possible evaluation could be asking

a different set of participants to draw the bar charts that we used in this study by reading their summaries produced by an appropriate baseline approach, and comparing the scores that those new set of drawings received from the same three evaluators with our current results. This evaluation will also allow us to determine whether the evaluators judged our summarization approach favorably because they were aware of the overall approach (i.e., brief summaries are generated by our system and these summaries do not contain everything that can be conveyed by the corresponding bar charts).

Favorable results were also achieved when people with visual impairments were presented with the brief summaries generated by our work in an interactive system which also enabled the user to ask follow-up questions to learn more about about the graphic. More on that system and study can be found in Demir et al. (2010).

9. Conclusion and Future Work

The majority of information graphics from popular media are intended to convey a message that is often not captured by the text of the document. Thus graphics, along with the textual segments, contribute to the overall purpose of a multimodal document and cannot be ignored. The work presented in this article is the first to apply natural language generation technology to provide the message and high-level knowledge that one would gain by viewing graphics from popular media via brief textual summaries. Our summarization approach treats a graphic as a form of language and utilizes the inferred intention of the graphic designer, the communicative signals present in the graphic, and the significant visual features of the underlying data in determining what to convey about that graph. Our approach uses a set of content identification rules constructed for each intended message category of a bar chart in determining the content of the summaries. The propositions selected for inclusion by these rules are organized into a text structure by applying a novel bottom–up approach which leverages different discourse related considerations such as the number and syntactic complexity of sentences and clause embeddings that will be used for realization. Following the generation of referring expressions for certain graphical elements, the structure of a summary is realized in natural language via a publicly available realizer. Three different evaluation studies validated the effectiveness of our approach in selecting and coherently organizing the most important information that should be conveyed about a bar chart, and enabling readers to correctly understand the high-level knowledge of the graphic.

In addition to the application area, this article makes contributions to two broad areas of research: data-to-text generation systems and text-to-text generation of referring expressions. Here we have viewed the generation of a summary of an information graphic as a data-to-text generation problem. Any data-to-text generation system must solve several important problems: (1) out of all of the information in the data, extract out that information that is important enough to be included in the text, (2) structure the information so it can be realized as a coherent text, (3) aggregate propositions to be conveyed in the text into more complex yet understandable sentence structures, (4) order the resulting sentence structures so as to maintain text fluency, and (5) realize the information as English sentences and generate appropriate referring expressions. Our work has addressed each of these issues in a systematic fashion maintaining modularity of system components and following a development methodology that includes human input for making system decisions, and a thorough evaluation of each module as well as final system evaluation.

Although the specific implementations that we developed are geared toward generating summaries of bar charts, the groundwork described in this article is currently

being used by our own group to investigate summarizing other types of graphics (such as line graphs and grouped bar charts) from popular media. A Bayesian system for recognizing the intended message of line graphs has already been developed (Wu et al. 2010) and the work on constructing the content identification rules for line graphs is under way (Greenbacker, Carberry, and McCoy 2011).

The module that generates referring expressions represents a sophisticated study of text-to-text referring expression generation. Referring expression generation is a vibrant field. Although the particular rules used for extracting an appropriate referring expression are unique to referring expressions for graphical elements inside an information graphic (note: not just bar charts), the work has uncovered some rather interesting properties in terms of extracting expressions from text which may generalize to other domains as well. Our work is unique in that a full referring expression is pieced together from text not directly referring to the item in question. In order to do this, we identified text levels and rules for extracting the core expression along with potential important modifiers. One can imagine this message carrying over to other types of data-to-text activities as well as to more standard text-to-text generation problems.

In future work, we intend to build on the work reported here in several ways. Corpus studies where human subjects tasked with writing brief summaries of graphics (the kind of summaries that our system intends to generate) would be of great potential in informing generation decisions that our system makes at different levels. Moreover, experts with extensive knowledge of the domain or the targeted end users were shown to be of greater supply to the development of many NLG systems. Learning from summaries written by subjects (especially expert writers) would be an exciting area of future research. We also believe that our approach would benefit from these corpus studies towards exploring how the fluency of the summaries can be further improved particularly by reducing the occasional verbosity in order to achieve textual economy (Stone and Webber 1998). For example, these efforts might help us to determine how measurement axis descriptors can be more appropriately phrased in different situations. Improving the current evaluation metric for choosing a text plan is also in our research agenda. We utilize three criteria in our evaluation metric for determining the best structure that can be obtained by applying the operators to the selected propositions. In addition, each criteria (the number of sentences, the overall syntactic complexity of sentences, and the overall comprehension complexity of all embedded clauses) has the same impact on the selection process. We are considering conducting more user studies in order to identify what other criteria should be taken into account and how important each criteria should be in relation to all the others. Moreover, exploring the broader applicability of the novel aspects of our work in other settings is an interesting topic for future work. Finally, evaluating the utility of our theoretically informed aggregation mechanism in comparison to or in conjunction with the more surface-oriented mechanism of the SPoT system would be a promising area for further research.

To our best knowledge, what kinds of summaries best serve the needs of visually impaired individuals has not been throughly studied before. As mentioned in Section 2.2.1, we believe that our summaries, once associated with graphics as ALT texts, might be of help to these individuals while reading electronic documents via screen readers. One fruitful research direction would be to present such individuals with our summaries in real-time scenarios and to mine their informational and presentational needs. Such a study would probably provide insights with regard to this question and potentially lead to guidelines that human summarizers could follow in generating summaries for people with visual impairments.

Acknowledgments

The authors would like to thank the study volunteers for their time and willingness to participate. This material is based upon work supported by the National Institute on Disability and Rehabilitation Research under grant no. H133G080047. For all graphics that were used in our evaluations and are given as examples in this article, inputs (i.e., the inferred intended message and the XML representation of the graphic) and outputs (i.e., the textual summary of the graphic) of our system can be found at the following Web page: `www.cis.udel.edu/~carberry/barcharts-corpus/`.

References

Alty, James L. and Dimitrios I. Rigas. 1998. Communicating graphical information to blind users using music: the role of context. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 574–581, Los Angeles, CA.

Baldwin, Breck and Thomas Morton. 1998. Dynamic coreference-based summarization. In *Proceedings of the 3rd Conference on Empirical Methods in Natural Language Processing*, pages 1–6, Granada.

Barzilay, Regina, Noemie Elhadad, and Kathleen McKeown. 2002. Inferring strategies for sentence ordering in multidocument news summarization. *Journal of Artificial Intelligence Research*, 17:35–55.

Barzilay, Regina and Mirella Lapata. 2006. Aggregation via set partitioning for natural language generation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 359–366, New York, NY.

Belz, Anja, Eric Kow, Jette Viethen, and Albert Gatt. 2008. The Grec challenge 2008: Overview and evaluation results. In *Proceedings of the 5th International Natural Language Generation Conference*, pages 183–191, Salt Fork, OH.

Brennan, Susan E., Marilyn W. Friedman, and Carl J. Pollard. 1987. A centering approach to pronouns. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics*, pages 155–162, Stanford, CA.

Carberry, Sandra, Stephanie Elzer, and Seniz Demir. 2006. Information graphics: An untapped resource for digital libraries. In *Proceedings of the ACM Special Interest Group on Information Retrieval Conference*, pages 581–588, Seattle, WA.

Chester, Daniel and Stephanie Elzer. 2005. Getting computers to see information graphics so users do not have to. In *Proceedings of the 15th International Symposium on Methodologies for Intelligent Systems*, pages 660–668, Saratoga Springs, NY.

Clark, Herbert. 1996. *Using Language*. Cambridge University Press, Cambridge.

Coch, Jose. 1998. Interactive generation and knowledge administration in multimeteo. In *Proceedings of 9th International Workshop on Natural Language Generation*, pages 300–303, Niagara-on-the-Lake.

Corio, Marc and Guy Lapalme. 1999. Generation of texts for information graphics. In *Proceedings of the 7th European Workshop on Natural Language Generation*, pages 49–58, Toulouse.

Covington, M., C. He, C. Brown, L. Naci, and J. Brown. 2006. How complex is that sentence? A proposed revision of the rosenberg and abbeduto D-level scale. Research Report, Artificial Intelligence Center, University of Georgia.

Cycorp. Open Cyc. 2011. `http://www.cyc.com`.

Dale, Robert and Ehud Reiter. 1995. Computational interpretations of the gricean maxims in the generation of referring expressions. *Cognitive Science*, 19(2):233–263.

Dalianis, Hercules. 1999. Aggregation in natural language generation. *Computational Intelligence*, 15(4):384–414.

Demir, Seniz, Sandra Carberry, and Stephanie Elzer. 2009. *Issues in Realizing the Overall Message of a Bar Chart*, John Benjamins, 5th edition. Amsterdam, pages 311–320.

Demir, Seniz, David Oliver, Edward Schwartz, Stephanie Elzer, Sandra Carberry, Kathleen F. McCoy, and Daniel Chester. 2010. Interactive sight: Textual access to simple bar charts. *The New Review of Hypermedia and Multimedia*, 16(3):245–279.

Di Eugenio, Barbara, Davide Fossati, Dan Yu, Susan Haller, and Michael Glass. 2005. Aggregation improves learning: Experiments in natural language generation for intelligent tutoring systems. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 50–57, Ann Arbor, MI.

Elhadad, M. and J. Robin. 1999. SURGE: A comprehensive plug-in syntactic realization component for text generation. Technical Report, Department of Computer Science, Ben Gurion University. Beersheba, Israel.

Elzer, Stephanie, Sandra Carberry, and Ingrid Zukerman. 2011. The automated understanding of simple bar charts. *Artificial Intelligence*, 175(2):526–555.

Elzer, Stephanie, Nancy Green, Sandra Carberry, and James Hoffman. 2006. A model of perceptual task effort for bar charts and its role in recognizing intention. *International Journal on User Modeling and User-Adapted Interaction*, 16(1):1–30.

Fasciano, Massimo and Guy Lapalme. 2000. Intentions in the coordinated generation of graphics and text from tabular data. *Knowledge and Information Systems*, 2(3):310–339.

Fellbaum, Christiane. 1998. *WordNet: An Electronic Lexical Database*. The MIT Press, Cambridge, MA.

Ferres, Leo, Petro Verkhogliad, Gitte Lindgaard, Louis Boucher, Antoine Chretien, and Martin Lachance. 2007. Improving accessibility to statistical graphs: the igraph-lite system. In *Proceedings of the 9th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 67–74, Tempe, AZ.

Foster, Mary Ellen. 1999. Automatically generating text to accompany information graphics. Master's Thesis, University of Toronto.

Friendly, Michael. 2008. A brief history of data visualization. In C. Chen, W. Härdle, and A. Unwin, editors, *Handbook of Computational Statistics: Data Visualization*, volume III. Springer-Verlag, Heidelberg, pages 1–34.

Gatt, Albert, Francois Portet, Ehud Reiter, Jim Hunter, Saad Mahamood, Wendy Moncur, and Somayajulu Sripada. 2009. From data to text in the neonatal intensive care unit: Using NLG technology for decision support and information management. *AI Communications*, 22(3):153–186.

Goldberg, Eli, Norbert Driedger, and Richard I. Kittredge. 1994. Using natural-language processing to produce weather forecasts. *IEEE Expert: Intelligent Systems and Their Applications*, 9(2):45–53.

Goldstein, Jade, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. 2000. Multi-document summarization by sentence extraction. In *Proceedings of the NAACL-ANLP Workshop on Automatic Summarization*, pages 40–48, Seattle, WA.

Greenbacker, Charlie, Sandra Carberry, and Kathleen F. McCoy. 2011. A corpus of human-written summaries of line graphs. In *Proceedings of the EMNLP 2011 Workshop on Language Generation and Evaluation (UCNLG+Eval)*, pages 23–27, Edinburgh.

Grice, H. Paul. 1975. Logic and conversation. *Speech Acts*, 3:41–58.

Grosz, Barbara and Candace Sidner. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Grosz, Barbara J., Scott Weinstein, and Aravind K. Joshi. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225.

Hovy, Eduard H. 1988. Planning coherent multisentential text. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics*, pages 163–169, Buffalo, NY.

Hovy, Eduard H. 1993. Automated discourse generation using discourse structure relations. *Artificial Intelligence*, 63(1-2):341–385.

Hovy, Eduard and Chin-Yew Lin. 1996. Automated text summarization and the summarist system. In *Proceedings of the Workshop on TIPSTER Text Program*, pages 197–214, Vienna, VA.

Ina, Satoshi. 1996. Computer graphics for the blind. *SIGCAPH Computers and the Physically Handicapped*, 55:16–23.

Jayant, Chandrika, Matt Renzelmann, Dana Wen, Satria Krisnandi, Richard Ladner, and Dan Comden. 2007. Automated tactile graphics translation: in the field. In *Proceedings of the 9th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 75–82, Tempe, AZ.

Johnson, Mark. 1998. Proof nets and the complexity of processing center embedded constructions. *Journal of Logic, Language and Information*, 7(4):433–447.

Joshi, Aravind, Bonnie Webber, and Ralph Weischedel. 1984. Living up to expectations: Computing expert responses. In *Proceedings of the National Conference on Artificial Intelligence*, pages 169–175, Austin, TX.

Karamanis, Nikiforos, Chris Mellish, Massimo Poesio, and Jon Oberlander. 2009. Evaluating centering for information

ordering using corpora. *Computational Linguistics*, 35(1):29–46.
Kennel, A. 1996. Audiograf: A diagram-reader for the blind. In *Proceedings of the 2nd Annual ACM Conference on Assistive Technologies*, pages 51–56, Vancover, BC, Canada.
Kerpedjiev, Stephan and Steven Roth. 2000. Mapping communicative goals into conceptual tasks to generate graphics in discourse. In *Proceedings of the International Conference on Intelligent User Interfaces*, pages 60–67, New Orleans, LA.
Kibble, Rodger and Richard Power. 2004. Optimizing referential coherence in text generation. *Computational Linguistics*, 30(4):401–416.
Kidd, Evan and Edith Bavin. 2002. English-speaking children's comprehension of relative clauses: Evidence for general-cognitive and language-specific constraints on development. *Journal of Psycholinguistic Research*, 31(6):599–617.
Krahmer, E. and M. Theune. 2002. Efficient context-sensitive generation of referring expressions. In K. van Deemter and R. Kibble, editors, *Information Sharing: Reference and Presupposition in Language Generation and Interpretation, Center for the Study of Language and Information-Lecture Notes*, volume 143 of *CSLI Lecture Notes*. CSLI Publications, Stanford, CA, pages 233–264.
Krahmer, Emiel, Sebastiaan Van Erk, and André Verleg. 2003. Graph-based generation of referring expressions. *Computational Linguistics*, 29(1):53–72.
Kukich, Karen. 1983. Design of a knowledge-based report generator. In *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, pages 145–150, Cambridge, MA.
Kurze, Martin. 1995. Giving blind people access to graphics (example: business graphics). In *Proceedings of the Software-Ergonomie Workshop*, Dannstadt, Bremen.
Lavoie, Benoit and Owen Rambow. 1997. A fast and portable realizer for text generation systems. In *Proceedings of the 5th Conference on Applied Natural Language Processing*, pages 265–268, Washington, DC.
Lazar, J., A. Allen, J. Kleinman, and C. Malarkey. 2007. What frustrates screen reader users on the web: A study of 100 blind users. *International Journal of Human-Computer Interaction*, 22(3):247–269.
Lester, James C. and Bruce W. Porter. 1997. Developing and empirically evaluating robust explanation generators: The KNIGHT experiments. *Computational Linguistics*, 23(1):65–101.
Lin, Dekang. 1996. On the structural complexity of natural language sentences. In *Proceedings of the International Conference on Computational Linguistics*, pages 729–733, Copenhagen, Denmark.
Mann, William C. and Sandra A. Thompson. 1987. Rhetorical structure theory: A theory of text organization. In Livia Polanyi, editor, *The Structure of Discourse*. Ablex Publishing Corporation, Norwood, NJ.
Marcu, Daniel. 1998. *The Rhetorical Parsing, Summarization, and Generation of Natural Language Texts*. Ph.D. thesis, Department of Computer Science, University of Toronto.
McCoy, Kathleen F., Sandra Carberry, Tom Roper, and Nancy Green. 2001. Towards generating textual summaries of graphs. In *Proceedings of the 1st International Conference on Universal Access in Human-Computer Interaction*, pages 695–699, New Orleans, LA.
McCoy, Kathleen F. and Jeannette Cheng. 1991. Focus of attention: Constraining what can be said next. In Cecile Paris, William Swartout, and William Mann, editors, *Natural Language Generation in Artificial Intelligence and Computational Linguistics*. Kluwer Academic Publishers, Berlin, pages 103–124.
McKeown, Kathleen R. 1985. Discourse strategies for generating natural-language text. *Artificial Intelligence*, 27(1):1–41.
McKeown, Kathleen R., Shimei Pan, James Shaw, Desmond A. Jordan, and Barry A. Allen. 1997. Language generation for multimedia healthcare briefings. In *Proceedings of the 5th Conference on Applied Natural Language Processing*, pages 277–282, Washington, DC.
Meijer, Peter B. 1992. An experimental system for auditory image representations. *IEEE Transactions on Biomedical Engineering*, 39(2):112–121.
Mellish, Chris, Alisdair Knott, Jon Oberlander, and Mick O'Donnell. 1998. Experiments using stochastic search for text planning. In *Proceedings of the 9th International Workshop on Natural Language Generation*, pages 98–107, Niagara-on-the-Lake.
Moore, Johanna D. and Cecile Paris. 1993. Planning text for advisory

dialogues: Capturing intentional and rhetorical information. *Computational Linguistics*, 19(4):651–694.
Nenkova, Ani and Kathleen McKeown. 2003. References to named entities: a corpus study. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 70–72, Edmonton.
O'Donnell, M., C. Mellish, J. Oberlander, and A. Knott. 2001. Ilex: an architecture for a dynamic hypertext generation system. *Natural Language Engineering*, 7(3):225–250.
Paris, Cecile. 1988. Tailoring object descriptions to a user's level of expertise. *Computational Linguistics*, 14(3):64–78.
Pastra, Katerina, Horacio Saggion, and Yorick Wilks. 2003. Extracting relational facts for indexing and retrieval of crime-scene photographs. *Knowledge-Based Systems*, 16(5-6):313–320.
Portet, Francois, Ehud Reiter, Albert Gatt, Jim Hunter, Somayajulu Sripada, Yvonne Freer, and Cindy Sykes. 2009. Automatic generation of textual summaries from neonatal intensive care data. *Artificial Intelligence*, 173(7-8):789–816.
Radev, Dragomir R., Hongyan Jing, Malgorzata Stys, and Daniel Tam. 2004. Centroid-based summarization of multiple documents. *Information Processing and Management: An International Journal*, 40(6):919–938.
Ramloll, Rameshsharma, Wai Yu, Stephen Brewster, Beate Riedel, Mike Burton, and Gisela Dimigen. 2000. Constructing sonified haptic line graphs for the blind student: First steps. In *Proceedings of the 4th International ACM Conference on Assistive Technologies*, pages 17–25, Arlington, VA.
Reiter, Ehud. 2007. An architecture for data-to-text systems. In *Proceedings of the 11th European Workshop on Natural Language Generation*, pages 97–104, Schloss Dagstuhl.
Reiter, Ehud and Robert Dale. 2000. *Building Natural-language Generation Systems*. Cambridge University Press, Cambridge.
Schiffman, Barry, Ani Nenkova, and Kathleen McKeown. 2002. Experiments in multidocument summarization. In *Proceedings of the 2nd International Conference on Human Language Technology Research*, pages 52–58, San Diego, CA.
Shaw, James. 1998. Clause aggregation using linguistics knowledge. In *Proceedings of the 9th International Workshop on Natural Language Generation*, pages 138–147, Niagara-on-the-Lake.
Somayajulu, Sripada, Ehud Reiter, and Ian Davy. 2003. Sumtime-mousam: Configurable marine weather forecast generator. *Expert Update*, 6(3):4–10.
Stent, A., Rashmi Prasad, and Marilyn Walker. 2004. Trainable sentence planning for complex information presentation in spoken dialog systems. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, pages 79–86, Barcelona.
Stone, Matthew and Bonnie Webber. 1998. Textual economy through closely coupled syntax and semantics. In *Proceedings of the International Natural Language Generation Conference*, pages 178–187, Niagara-on-the-Lake.
Suri, Linda Z. and Kathleen F. McCoy. 1994. Raft/rapr and centering: A comparison and discussion of problems related to processing complex sentences. *Computational Linguistics*, 20(2):301–317.
Turner, Ross, Yaji Sripada, and Ehud Reiter. 2009. Generating approximate geographic descriptions. In *Proceedings of the 12th European Workshop on Natural Language Generation*, pages 42–49, Athens.
Walker, M., O. Rambow, and M. Rogati. 2002. Training a sentence planner for spoken dialogue using boosting. *Computer Speech and Language: Special Issue on Spoken Language Generation*, 16(3):409–434.
Walker, M., A. Stent, F. Mairesse, and R. Prasad. 2007. Individual and domain adaptation in sentence planning for dialogue. *Journal of Artificial Intelligence Research*, 30(1):413–456.
Wu, Peng, Sandra Carberry, Stephanie Elzer, and Daniel Chester. 2010. Recognizing the intended message of line graphs. In *Proceedings of the International Conference on the Theory and Application of Diagrams*, pages 220–234, Portland, OR.
Yngve, Victor H. 1960. A model and an hypothesis for language structure. *American Philosophical Society*, 104:444–466.
Yu, Jin, Ehud Reiter, Jim Hunter, and Chris Mellish. 2007. Choosing the content of textual summaries of large time-series data sets. *Natural Engineering*, 13(1):25–49.